A MICROPROCESSOR BASED RAIL TRACK MONITORING SYSTEM

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

By
G. N. M. SUDHAKAR

to the
DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
SEPTEMBER, 1982

CERTIFICATE

Certified that the work entitled 'A MICROPROCESSOR BASED RAIL TRACK MONITORING SYSTEM' by G.N.M. Sudhakar has been carried out under our supervision and this has not been submitted elsewhere for a degree.

Dr. K.R. Sarma
Professor
Department of Elect. Engg.
I.I.T. Kanpur

Dr. V.P. Sinha
Professor
Department of Elect. Engg.
I.I.T. Kanpur

ACKNOWLEDGEMENTS

I would like to express my thanks to Dr. K.R. Sarma and Dr. V.P. Sinha for suggesting the topic of this thesis and extending the guidance and encouragement during the course of this work.

I am grateful to Mr. Bh. A.R.B. Raju, Research Engineer, for his timely help, useful discussions and comments during the design, fabrication part of the system hardware.

My special thanks go to Mr. B.V. Ramana, Research Engineer, I.I.T. Kanpur, for his help during the testing and assembling of different hardware modules. A few discussions with him helped me a lot in the software development.

I would also like to express my thanks to my friends Satyanarayana, Phani, Gopi, Sridhar and Satyaprasad for their help in proof reading, sketching etc.

My thanks are also due to Mr. J.D. Varma of Mechanical Engineering Department for his efficient typing of the thesis.

G.N.M. Sudhakar

MY SPECIAL THANKS GO TO SRI. DESH DEEPAK JOINT DIRECTOR, RDSO, LUCKNOW AND HIS COLLEAGUES FOR SUPPLYING THE NECESSARY INFORMATION AND DATA NEEDED FOR THIS WORK

G. N. M. SUDHAKAR

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Rail track monitoring is very essential to have a smooth and safe ride on railcars. Standards have been adopted to provide a minimum comfort level for a passenger. Hence it is necessary that a suitable track monitoring system should be available for the measurement of track irregularities.

At present midchord measurement scheme has been adopted by Indian Railways. This has got poor frequency response in the pass band of interest. In this thesis, a better system, which makes use of accelerometers to pick up the track irregularities is proposed. Since the accelerometers give second derivatives of track irregularities, suitable digital filtering techniques using Digital Incremental Computation (DIC) scheme have been developed.

Since track categorization will be done on the basis of power spectral measurement of the track irregularity, Fast Fourier Transform algorithms are used for the spectrum analysis.

A completely independent dedicated signal processor consisting of an input digitizer, a control unit using 6800 microprocessor, memory modules for both program and data storage, cassette, RS 232 C console and teletype interface modules, a small console module consisting of fifteen digit eight segment LEDs and a Hex

key board is developed; the system configuration and hardware details are described in detail.

The filtering and power spectral analysis is done on software basis on 6800. Assembly language programs are developed for this and are described in detail.

CHAPTER 1

INTRODUCTION

To have a smooth and safe ride of railway vehicles, the track tolerances should be within the acceptable limits and hence track monitoring is an essential part of railway operation. With the trend towards high speed transportation, the existing electrical and mechanical track irregularity measuring systems are inadequate and a better track monitoring system is proposed in this thesis.

In this chapter various parameters of the track irregularities are described and the methods of measurement are briefly reviewed. Also an outline of the proposed system is presented.

1.1 DEFINITION OF TRACK IRREGULARITIES

Traditionally the track irregularities have been described by the following parameters.

a) Unevenness of left rail
b) Unevenness of right rail
c) Alignment of left rail
d) Alignment of right rail
e) Cross level
f) Gauge
g) Twist

However, the track geometry can be equivalently described by the following four parameters (see Fig. 1.1)

a) Absolute centre line unevenness of the track - Z

b) Absolute centre line alignment of the track - Y

c) Cross level - XL

d) Gauge - G

Since the cross section of both rails change significantly during the service, it is necessary to define these parameters unambiguously. A typical track geometry is shown in Fig. 1.1. The various parameters are described below.

a) Centre Line Unevenness of Track (Z)

The centre line unevenness of track is defined as the variation of midpoint of track in vertical plane from an arbitrary datum. This datum will be a horizontal plane for tangent track and standard changing horizontal plane (taking into account super elevations) in case of curved tracks (Fig. 1.2)

b) Centre line Alignment of Track (Y)

The centre line alignment of track is defined as variation of midpoint of track in horizontal plane from an arbitrary datum. This datum is a straight line for tangent track and true circular or spiral shape in case of curved track (Fig. 1.3).

c) Cross Level (XL)

If the tangent of two rail head crowns make a slope with horizontal, the cross level is defined as 2L . ϕ where 2L is normal spacing between the centre line of two rail heads, and ϕ is the angle the tangent makes with horizontal datum as shown in Fig. 1.1.

d) Gauge (G)

Gauge is measured as the distance between the two rail heads some distance 'h' below the rail crown (h = 14 mm).

e) Midpoint of the Track (M)

The midpoint of the track is defined by first finding out midpoint of gauge line and then constructing a line perpendicular to the tangent of crowns of two rail head profiles.

The traditional parameters can be derived from these basic parameters as follows:

a) Unevenness of left rail will be $Z + \phi.L$

b) Unevenness of right rail will be $Z - \phi.L$

c) Alignment of left rail will be $Y + G/2$

d) Alignment of right rail will be $Y - G/2$

e) Twist will be $XL(1) - XL(2) / B$ where B = base; XL (1), XL (2) are the cross levels at the two end points of the base.

In the track irregularity measurements, the loaded or unloaded conditions play a very important role and these irregularities should be measured only under loaded conditions. Considering the present day speeds, the deflections of the track due to its own dynamics will be very small when compared to the loaded profile of the track geometry, hence are ignored. Also the deflections due to track springing are also ignored.

Since the moving railway vehicles will get the inputs from the track geometry irregularities and depending upon the speed of the vehicle these inputs correspondingly excite the vehicle which results in vibration of the vehicle components. The railway vehicles are strong resonant systems hence this interaction is best understood in frequency domain. Each track geometry parameter is modelled as a random process and by finding out the direct and cross spectras of the random processes a more useful information can be obtained in the design of track and Railway vehicles.

1.2 EXISTING MEASUREMENT SYSTEM

The Indian Railways at present monitor the routes using the Amsler track recording cars which mainly monitor the track geometry parameters like vertical unevenness, gauge and twist. Such cars measure these parameters through 'mid-chord versine measurement' using a 3.6 metre chord. The length is chosen to be compatible with the existing

bogie length. From the track geometry parameters obtained through the 3.6 metre chord, the tracks are classified into the categories A, B, C, D, depending upon the tolerances. However, the wavelength of the track defects was completely ignored in this analysis as shown below.

Let 'd' be the distance along the track and f(d) be the track irregularity. Let 'L' be the chord length used for the midchord measurement scheme. Let AB be the chord and C is the midpoint of the chord. Considering the irregularity to be a pure sine wave form, different cases are illustrated in Fig.1.4. From this it can be observed that the transfer function of the midchord at the 'peak inputs' can be approximated by

$$|H(\lambda)|^2 = \left|\left(1 - \cos\frac{\pi L}{\lambda}\right)\right|^2 \qquad (1.1)$$

where λ is the wavelength of the irregularity. However, as the chord moves away from the peak (Fig.1.4 case d), this approximation is not valid and the performance further deteriorates. Thus not only the value of λ but also the position of the chord affects the measurement of irregularity. Hence various frequencies present in the track irregularity are distorted by various amounts.

For a dynamic system like a railway vehicle, the response is completely dependent upon the input frequency. Since the midchord measurement vitiates the track geometry

by attenuating the longer wavelengths (since for large λ and for a given L, $L \ll \lambda$, the change in the irregularity is very small where as the irregularity with respect to a datum is not so), Amsler cars cannot measure the amplitudes of longer wavelengths.

It is a necessary condition that the wavelengths where the vehicles have substantial response should be least affected for the track irregularity measurements to have a relevance in estimating the vehicle response. Choosing such a chord which satisfies the above requirement for all vehicles at all speeds is a very difficult task. The speed variations come into picture because the vehicle frequencies in Hertz are related to the track irregularities through the speed of the vehicle, given by

$$f = \frac{v}{\lambda} \qquad (1.2)$$

where

v = speed of the vehicle in metres/sec.

λ = track irregularity wavelength in metres

f = frequency in Hz.

For higher speeds the larger wavelengths become more important and the chord length for midchord measurements has to be longer to ensure that track irregularities for long wavelengths are not attenuated.

A better system is proposed for the track irregularity measurement which makes use of accelerometers mounted on axle boxes for picking up the irregularity waveform. Since the accelerometers have good response in the wavelength range of interest a better estimation of the irregularities is possible.

1.3 PROPOSED SYSTEM

In the proposed system accelerometers are placed on axle boxes (Fig.1.5). Since the axle box is in contact with the rail wheels, the accelerometers pick the random vibration of the wheel caused by the track irregularity. These signals are suitably processed to obtain the actual irregularity through the use of suitable signal processing methods. The power spectra of these signals are evaluated for the purpose of track categorization and in the design of tracks and railway vehicles. The measurement of rail vertical profile and lateral profile is given below.

a) Absolute Vertical Profile

To measure the absolute vertical profile, an accelerometer is mounted on each of the 2 axle boxes of one wheel axle set of the vehicle. Assuming that the wheels will always remain in contact with the rail in vertical plane, the output of the accelerometer will be double integrated and suitably filtered in the wavelength band of 2 m to 40 m to acquire correct vertical profile. The central

line absolute vertical profile of the track, then, can be calculated by taking the average of the absolute vertical profile of left and right rail and the dynamic cross level will be evaluated by taking the difference of the absolute vertical profiles of left and right rails (Fig. 1.5).

b) Absolute Lateral Profile

This measurement is also similar to the above. But because of the standard play between the track gauge and wheel gauge, the assumption of wheel remaining in contact with the rail is no more valid in this case. Hence to measure absolute lateral profile, an accelerometer can be mounted on the axle box thus monitoring the lateral movement of the axle and distance of the inside rail face can be measured with respect to the axle with the help of two contact sensors, one each, looking into the rail face. Then the output of the two contact sensors and the lateral motions of the wheel set which will be obtained by double integrating will be manipulated to obtain the absolute lateral profile of left and right rail (Fig.1.6). Then the profiles are suitably filtered so that the correct information is available in the wavelength band of 2 m to 40 m and then by taking the average of lateral profile of left and right rail will give the central line alignment of the track and the difference gives the gauge of the track. Thus all the four basic parameters needed for the track geometry irregularity can be completely identified.

Data Processing

After acquiring all the four track geometry parameters namely central line unevenness, central line alignment, cross level and gauge, are to be suitably processed for using the same for track maintainence. For this processing, digital signal processing schemes will be much more useful, hence the data for every length of 512 points at an interval of 2 m is proposed for analysing purpose. The power spectral density of each of the track parameter will be evaluated for this 512 points length and the value of roughness constant will be calculated by fitting the following model of Power Spectral Density (PSD) for the track irregularities.

$$PSD = A \lambda^2 ; \quad PSD \text{ is in } mm^2/cycle/M ;$$

λ is wavelength in M

A is roughness constant.

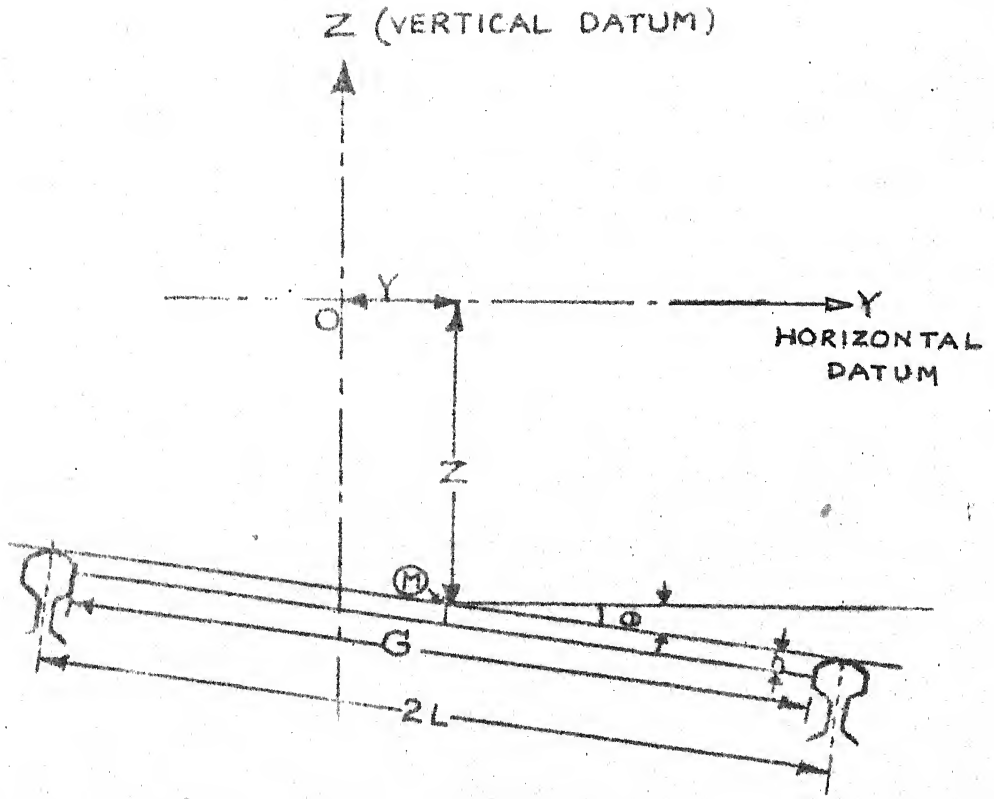
The value of A for each of the 4 track parameters can then be printed out.

1.4 PROPOSED SIGNAL PROCESSOR

As digital techniques offer various advantages over the analog schemes at these low frequencies, digital schemes are proposed in this system. For this, the data from the accelerometer is properly conditioned, digitized and then stored using appropriate signal conditioners and

Analog to Digital converters. The different operations are performed on the data to obtain the actual track irregularity. Although hardware digital filters can be used, as the power spectral measurement of the track irregularities is also involved, instead of building different dedicated hardware supports the whole system is configured using a completely independent dedicated microcomputer with one of the easily available microprocessors and its family support. In this way, all the digital filter schemes, and power spectral measurement of the track irregularities can be obtained through the software developed for the microcomputer.

6800 microprocessor and its family support is used in this system and a complete block diagram of the proposed system is shown in Fig. 4.1. The system consists of a central processing unit (CPU) module, memory modules for program and data storage, analog multiplexer and A/D converter modules for the sampling of input data and various interface modules to interact with the system. All these modules are assembled, tested and details are discussed in Chapter 4.



M - MID POINT OF TRACK

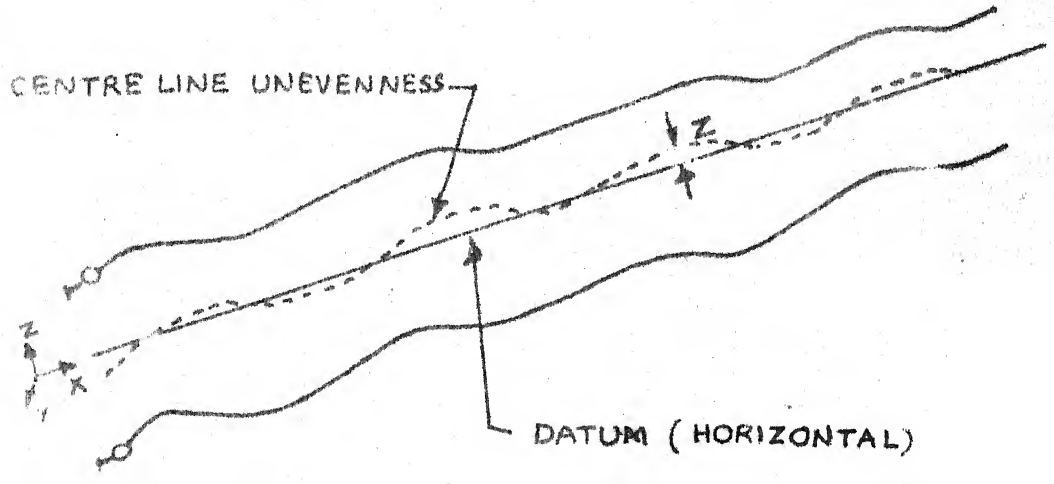
2L.ϕ - CROSS LEVEL

G - GAUGE

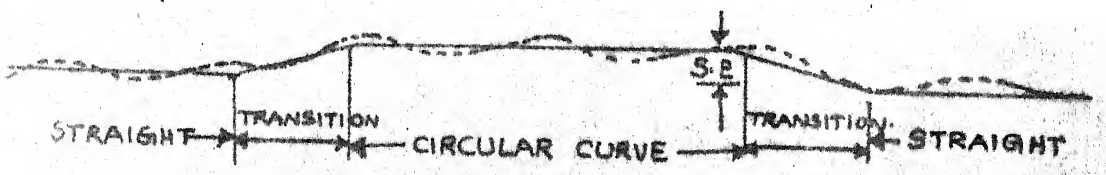
Z - CENTRE LINE UNEVENNESS

Y - CENTRE LINE ALIGNMENT

FIG. 1.1. DESCRIPTION OF TRACK IRREGULARITIES



a) STRAIGHT TRACK



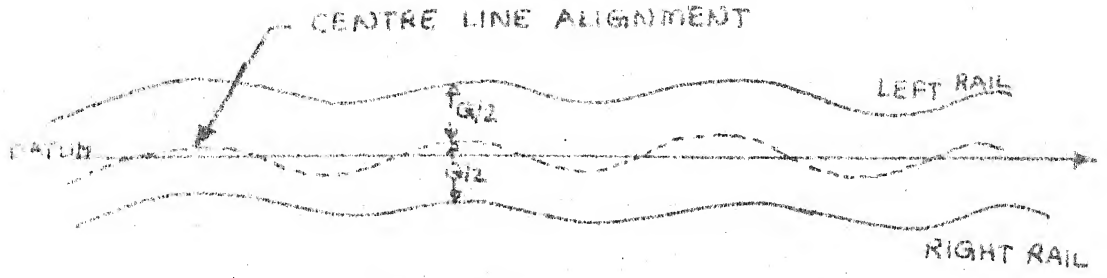
——— DATUM

– – – CENTRE LINE VERTICAL UNEVENNESS

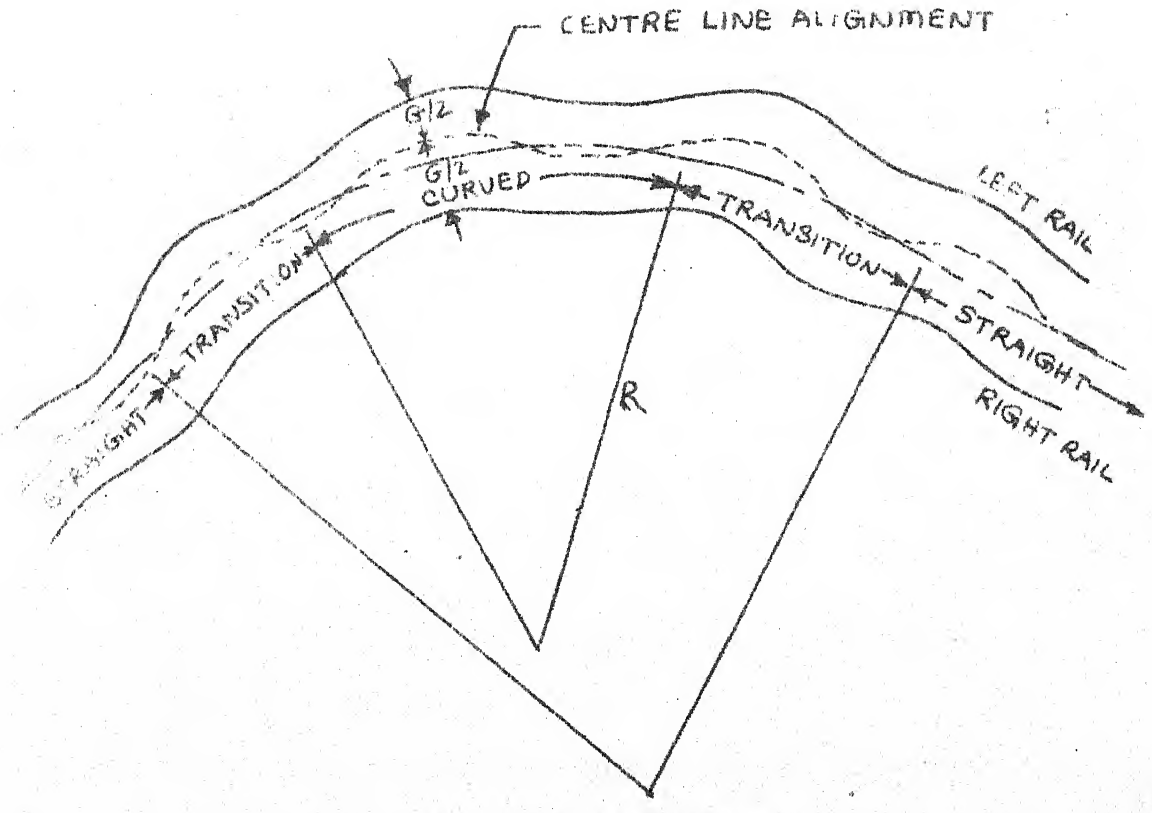
S.E SUPER ELEVATION

b) CURVED TRACK

FIG. 1.2. CENTRE LINE UNEVENNESS OF TRACK

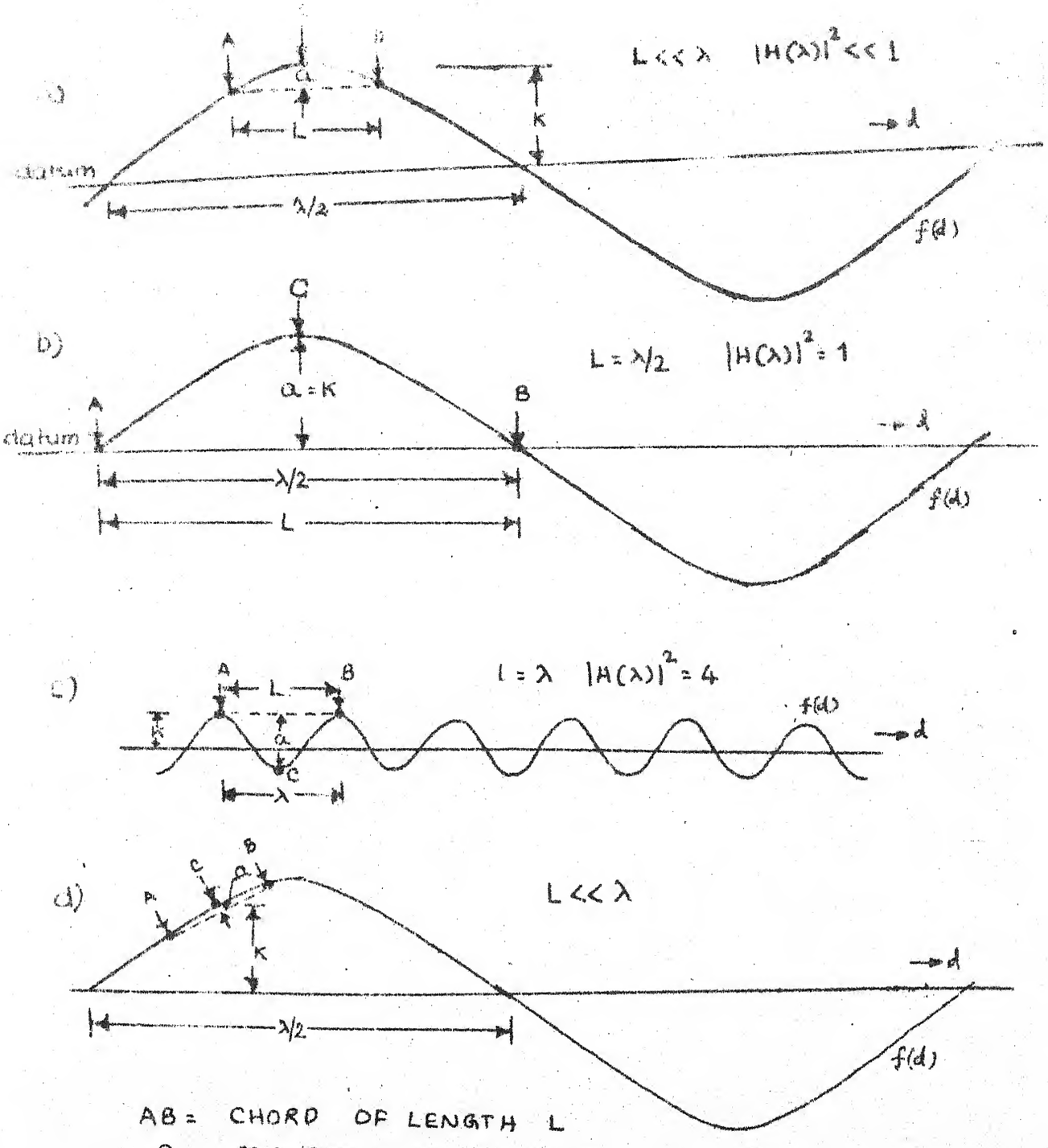


a) STRAIGHT TRACK



b) CURVED TRACK

FIG. 1.3. CENTRE LINE ALIGNMENT OF TRACK



AB = CHORD OF LENGTH L

C = MID POINT OF THE CHORD

f(d) = track irregularity

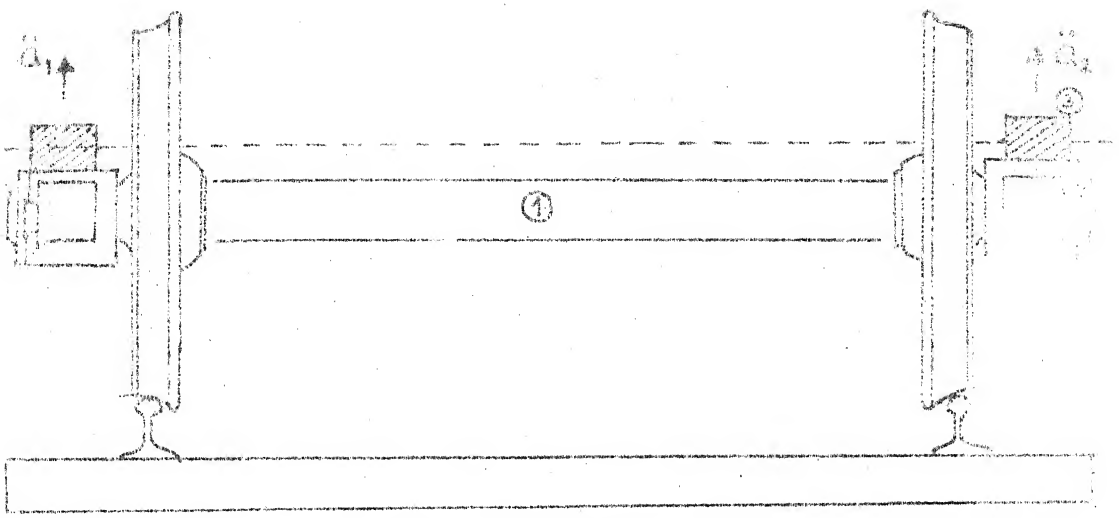
d = distance along the track

a = measured irregularity

K = True irregularity

λ = Wave length of the irregularity

FIG. 1.4. MIDCHORD MEASUREMENTS



① WHEEL AXLE SET
② AXLE BOX
③ ACCELOROMETER

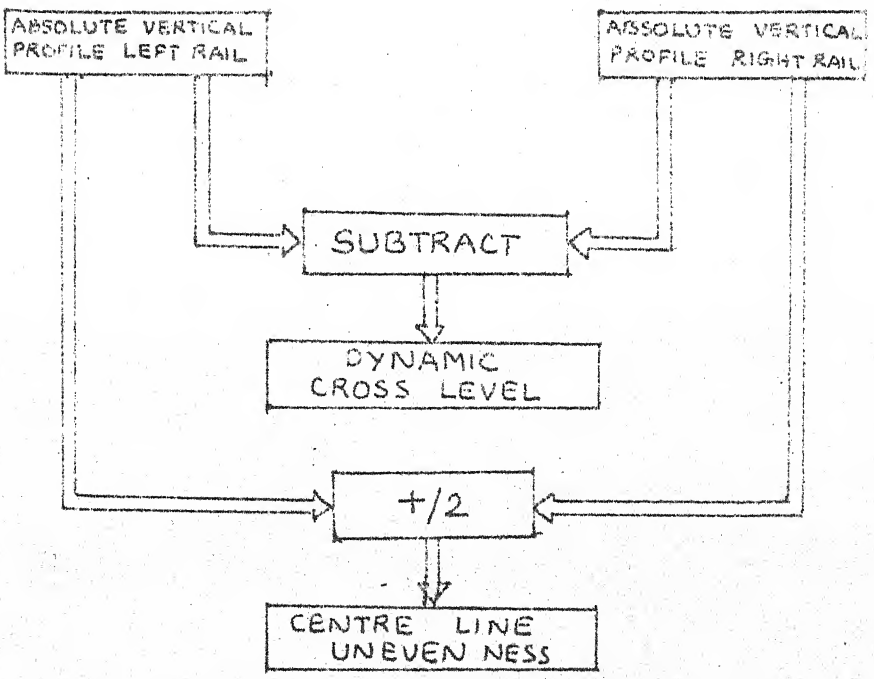
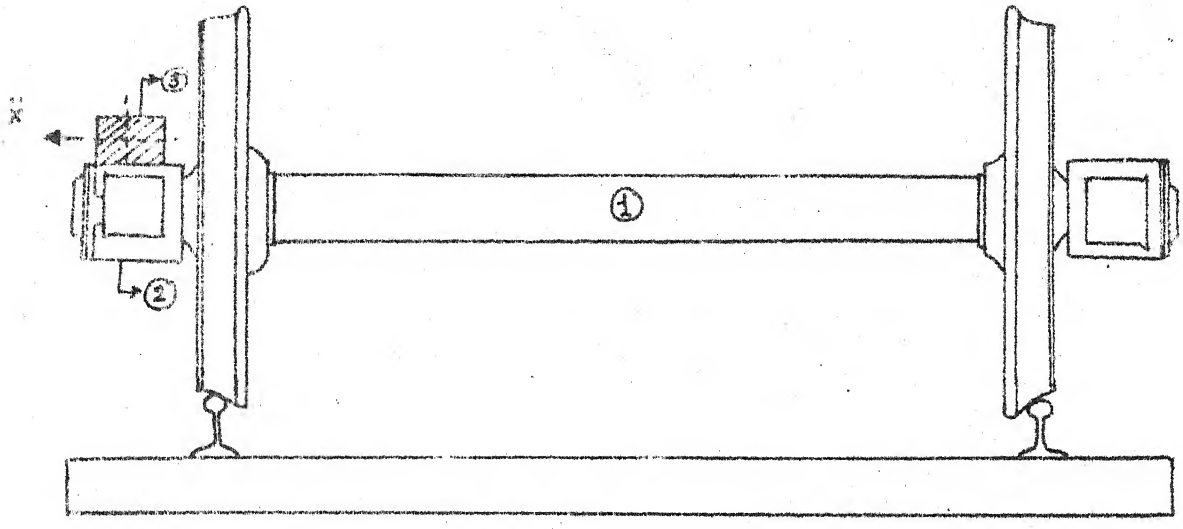


FIG. 1.5 ABSOLUTE VERTICAL PROFILE



1. WHEEL AXLE SET
2. AXLE BOX
3. ACCELEROMETER

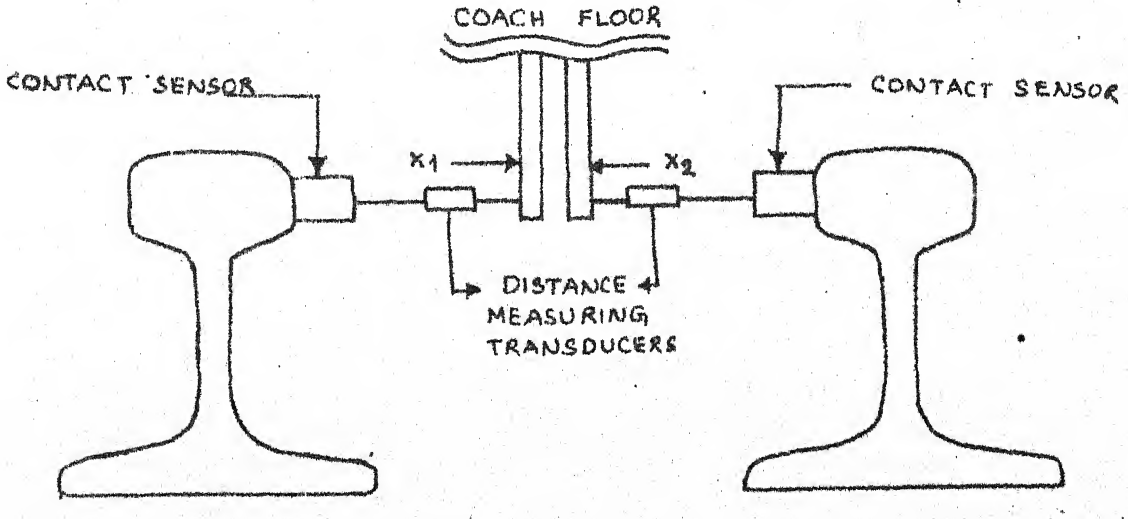
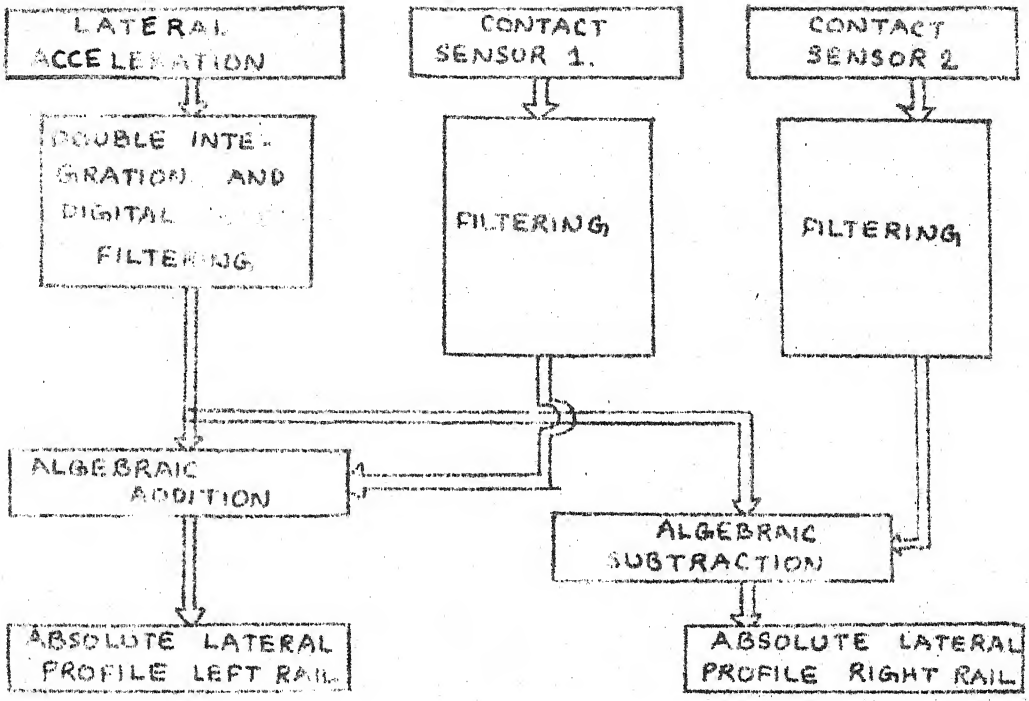
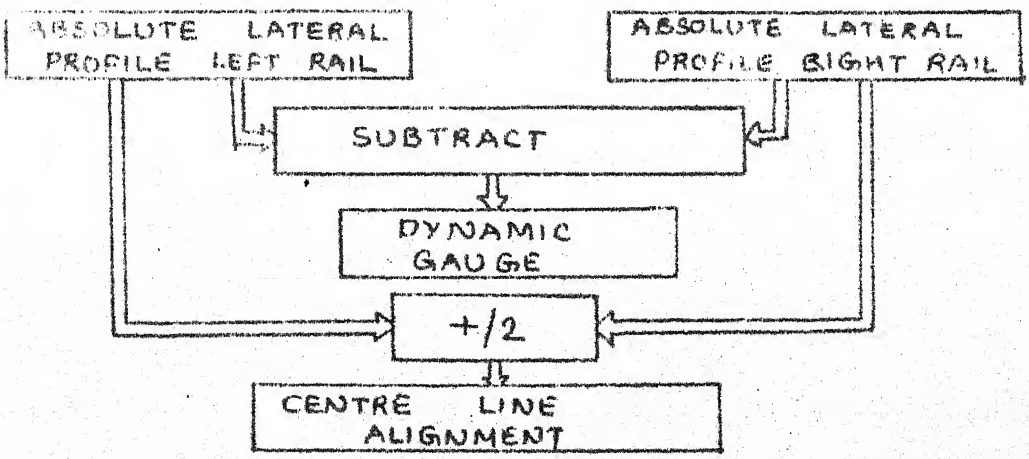


FIG. 1.6. LATERAL PROFILE MEASUREMENT





LATERAL PROFILE MEASUREMENT

CHAPTER 2

FILTERING TECHNIQUES FOR TRACK IRREGULARITY MEASUREMENT

2.1 INTRODUCTION

The proposed system for track monitoring measures the track irregularities using accelerometers mounted on axle boxes of the rail carriage. The accelerometers pick up the second derivatives of the track irregularities i.e., if a (d) is the irregularity, the output of the accelerometer is $\ddot{a}$ (d) where d is the distance along the track. There is thus a need for suitable filtering techniques to obtain the actual track irregularity from the accelerometer output signal.

Whether it is the measurement of vertical profile or lateral profile, an accelerometer is associated with each and therefore the method of obtaining the different irregularities will be the same viz., the output of the accelerometer is to be suitably conditioned and then double integrated, to obtain the actual irregularity a (d). Since the frequency of interest for the track irregularities is very low the design and stability of analog integrators at these frequencies pose various problems hence are not feasible for this application. A better approach will be to process by digital techniques in which the various inputs after sampling and storage are filtered using digital filters. These filters can be implemented either in hardware or software.

In the hardware scheme, the filters are implemented using arithmetic logic units, control and registers which are used for algebraic manipulation. In the software scheme, programs for various filters should be written which will then be processed using dedicated computers.

In this chapter, digital filter schemes are described which can be used for this application. Assembly language programs for 6800 microprocessor applications have been developed and are listed in Appendix.

2.2 DIGITAL INTEGRATORS

As mentioned, to obtain the track irregularity, the accelerometer output signal should be double integragrated. For adopting the digital schemes, these signals should be sampled and quantized using a proper Analog-to-Digital (A/D) converter. Since the variation in the track irregularity cannot be very large with respect to distance, the frequencies of interest are very small. Hence a simple Simpson Integration for this application can be adopted. This is shown in Fig. 2.1.

Let $f(x)$ be the accelerometer output; if $f(x)$ is sampled at a constant spatial interval of Δx, then

$$\int_{x=0}^{k} f(x)\, dx \simeq \sum_{i=0}^{k} f_{..}(x_i)\, \Delta x = \Delta x \sum_{i=0}^{k} f(x_i) \qquad (2.1)$$

where Δx is the sampling distance and $f(x_i)$ is the

sampled and quantized value of the signal at distance x_i. Since Δx, the sampling distance is constant, the integrator merely consists of an adder as given by Eq. (2.1), and a scaling factor.

If this scheme is directly applied to the digital computer applications, Integrator may overflow due to the limited precision of the computers. Especially when a DC component is overriding on the signal, this DC level will be adding up causing the computer to overflow. Similarly, depending upon the input signal, the output of the Integrator may consist a DC level thus causing overflow for the second integrator to be followed which is used for the double integration purpose. Similarly very low frequencies also cause the overflow of integrators. Hence a high pass prefilter which filters out all the DC and very low frequencies should be used before an integrator. A complete scheme for obtaining a track irregularity from the accelerometer is as shown in Fig. 2.2. In the following sections, the digital filtering schemes are described.

2.3 DIGITAL FILTERS [1], [2]

Analogus to Analog filters, there are many types of digital filters like Butterworth, Tschebycheff, Elliptical etc.[2]. The digital filters can also be represented using zero-pole plots as in the case of Analog filters. Considering the real time application of track

profile measurement, the computation time for the ditigal filter should not be very high. And also the magnitude of the signal in the pass band of interest should not be affected by the filter response. Hence a single pole high pass Butterworth filter is chosen for this application although higher order filters give much sharper cut-off frequency. Since the processing time for Non-Recursive filters [2] is high, the recursive filter implementation is the best possible solution for this application.

The transfer function of one-pole high-pass Butterworth filter in Z - domain is given by

$$\frac{Y(Z)}{X(Z)} = H(Z) = \frac{1 - Z^{-1}}{(w - 1) Z^{-1} + w + 1} \qquad (2.2)$$

where

Z = $\exp(2\pi i f/N)$

w = $\mathrm{Tan}(\pi f_c/N)$

N = Number of samples per second

f_c = Cut off frequency of the filter in cycles/sec.

f = any frequency for which transfer function is to be calculated.

From this equation, the recursive relation for the filter is

$$Y_i = \left(\frac{1}{1+w}\right)\left[(1-w)\,Y_{i-1} + X_i - X_{i-1}\right] \qquad (2.3)$$

where f_i represents the present value, f_{i-1} is one state earlier value. X_i represents the present input and Y_i is the present output. The above recursive equation can be realized as given in Fig. (2.3).

As it can be seen from this realization, since the present input is subtracted from the previous input, the DC component is removed at the output of the filter. Since w is a function of cut off frequency, by properly choosing a value for w, the cut off frequency can be fixed. Since all the computer operations will be in binary arithmetic, it is better to choose a 'w' which is in multiples of 2 so that divisions and multiplications can be performed very easily. In the practical track monitoring routines, it was observed that a cut off frequency of 0.125 Hz is suitable.Considering the present speed range of interest (120 KMPH) and sampling rate of 512 samples per every Km, a value of 2^{-5} for w is chosen.

For the integration, the trapezoidal integration scheme also works out to be better, for which the transfer function is given by

$$I(Z) = \frac{h}{2}\left(\frac{1+Z^{-1}}{1-Z^{-1}}\right) \qquad (2.4)$$

from which the recursive relation works out to be

$$Y_i = Y_{i-1} + \left(\frac{X_i + X_{i-1}}{2} \right) h \qquad (2.5)$$

where f_i is the present value, f_{i-1} is one state earlier value, h is the sampling interval. The recursive relation for high pass filter and a trapezoid integrator combined together will be

$$Y_i = \left(\frac{1}{1 + w} \right) \left[(1 - w)\, Y_{i-1} + \left(\frac{X_i + X_{i-1}}{2} \right) h \right] \qquad (2.6)$$

The composite filter is simulated on DEC 1090 computer. Whether it is the Simpson or trapezoidal integration , the following observations are made for the composite filter

a) Settling time of the composite filter is high.

b) When the complete double integration scheme is implemented, due to the settling time error, there is a fluctuating DC component observed at the output.

As this setting time is large, the filtered output doesn't correspond to the actual profile for this period. Hence, during this time the data is to be ignored. A better alternative in which settling time can be reduced is the 'incremental computation scheme' which works on the principle of differential input - output scheme in which increments in the data are used for processing.

2.4 INCREMENTAL COMPUTATION SCHEME FOR DIGITAL FILTERING OF LOW FREQUENCY SIGNALS [4]

Normally, for the digital filters which operate at low frequencies, the ratio of cut off to sampling frequency is very small. In other words, for the filters which have low ratios of cut-off to sampling frequency, the poles are located close to the unit circle in the Z - plane. The recursive filters which operate with the poles very close to Z = 1 and use finite word lengths are prone to the effects of sensitivity and round off errors. A great deal of work has been done to reduce these errors and also different schemes have been presented in using the incremental computation for the low frequency signals. Detailed discussions can be found in the reference [4].

2.4.1 Difference Between Normal and Incremental Computation [4]

Definition:

a) Normal Computation : A computation procedure that evaluates each value the dependent variable assumes independent of its previous values is referred to as the normal computation scheme. The value of the dependent variable computed in this manner is termed 'normal value'.

b) Incremental Computation: Normally many physical system are slowly varying. In other words, for a given change in the independent variable, the amount by which the dependent variable changes from its previous value is very small. The succeeding value that the dependent variable assumes can be computed by adding an increment or decrement to the one previously computed. A scheme that employs this principle in evaluating values that the dependent variable assumes is referred to as an incremental computation.

2.4.2 Using Digital Incremental Computation (DIC) for Digital Filtering

Let the required transfer function be $H(Z) = Y(Z) / X(Z)$. In the conventional filters where the normal scheme of computation is adopted, the input and output are not in incremental form. In the DIC scheme the input is followed by a differencer, a transfer function $H'(Z)$ for which the input and output are in incremental form and an output integrator as shown in Fig. 2.4. As from this illustration, for the two structures to be identical the transfer functions $H(Z)$ and $H'(Z)$ should be identical. A simple DIC structure of first order filter is shown in Fig. 2.5. $e_1(n)$ and $e_2(n)$ are the noise sources arising from the quantization operations for multiplication and integration. The transfer function $H'(Z)$ of the DIC structure is

$$H'(Z) = \frac{Y(Z)}{X(Z)} = \frac{(B_o + B_1 G_1) - B_o Z^{-1}}{1 - (1 - A_1 G_1) Z^{-1}} \qquad (2.7)$$

The transfer function of the first order conventional filter is

$$H(Z) = \frac{Y(Z)}{X(Z)} = \frac{c - d Z^{-1}}{1 - a Z^{-1}} \qquad (2.8)$$

In order to the transfer functions to be equivalent corresponding coefficients in transfer functions are to be equated. From this, the DIC filter coefficients are given by

$$\begin{aligned} B_o &= d \\ B_1 &= (c - d) / G_1 \\ A_1 &= (1 - a) / G_1 \end{aligned} \qquad (2.9)$$

where G_1 is the gain of the integrator.

A block diagram of the DIC filter with inputs and outputs in normal form is shown in Fig. 2.6. The coefficients A_i, B_i will be chosen such that $H'(Z) = G_o H(Z)$ where G_o is a scale factor and is imposed by the structure since it operates upon the increments of the input. And also G_o can be used for controlling the dynamic range of the filter. Normally G_o should be chosen such that $|\Delta Y|_{max} = |\Delta X|_{max}$. In the above DIC structured filter, the input and output are in incremental form. In order to specify the complete structure of the DIC, the scheme in Fig. 2.6 will be adopted where input and output are in normal form.

Second Order Filter

A second order filter can also be derived using DIC schemes on similar lines. Consider the second order transfer function

$$H(Z) = \frac{c - d\,Z^{-1} + Z^{-2}}{1 - a\,Z^{-1} + b\,Z^{-2}} \qquad (2.10)$$

The structures of second order conventional and DIC filters are given in Fig. 2.7 and Fig. 2.8. From Fig. 2.8 the transfer function of DIC filter is given by

$$H'(Z) = \frac{(B_o - B_1 G_2 + B_2 G_1 G_2) - (2 B_o + B_1 G_2) Z^{-1} + B_o Z^{-2}}{1 - (2 - A_1 G_2 - A_2 G_1 G_2) Z^{-1} + (1 - A_1 G_2) Z^{-2}} \qquad (2.11)$$

The corresponding coefficients are

$B_o = 1$

$B_1 = (d - 2) / G_2$

$B_2 = (1 + c - d) / G_1 G_2$

$A_1 = (1 - b) / G_2$

$A_2 = (1 - a + b) / G_1 G_2$

where $G_i = 2^{-b_i}$ is the gain of the integrator. $e_1(n)$, $e_2(n)$, $e_3(n)$ are the noise sources arising from the quantization operation of multiplication and integration.

2.5 NOISE STUDY, DESIGN PROCEDURE [4]

The roundoff errors that occur in recursive digital filters employing fixed point arithmetic are mainly due to the finite world length of the digital computers.

The first error is caused by the addition of two fixed-point numbers resulting in overflow. By restricting the dynamic range of the signal level, the overflow can be controlled not only at the input and output but also at other intermediate points. The second error is due to the multiplication of the signal with the filter coefficients. When two numbers are multiplied, the resulting product word length is not equal to the sum of the lengths of the two words being multiplied. In a recursive filter, this product is again multiplied with a sample of the input signal in the next recursion. But the length of the product should be quantized to limit the register length. This quantization of the product word gives rise to an error that accumulates as succeeding multiplications are made. The model for fixed point round off noise following a multiplication is shown in Fig. 2.9.

2.5.1 Noise Power in Second Order Recursive Filters

Consider a transfer function for second order recursive filter

$$H(Z) = \frac{c}{1 - a Z^{-1} + b Z^{-2}} \qquad (2.13)$$

For obtaining the model of recursive filter the following assumptions are made. The operation of quantization used is rounding.

a) Any two different samples from the same noise sources are uncorrelated.

b) Any two noise sources associated with different multipliers are uncorrelated.

c) Each noise source is uncorrelated with the input sequence.

Let the variable be represented by (b' + 1) bits in the fixed point arithmetic which includes the sign bit also. Then each noise source can be modelled as a discrete stationary white random process with uniform power spectral density as shown in Fig. 2.10.

With the above assumptions the filter can be modelled as shown in Fig. 2.11. When the input signal is completely absent or becomes constant, the three assumptions of uncorrelatedness fail and the filter exhibits an oscillatory behaviour at the output which is termed LIMIT CYCLE OSCILLATION. The round off noise power at the output is given by [4]

$$\sigma_c^2 = \frac{1 + b}{6\,(1 - b)\,(1 + b - a)\,(1 + b + a)} \qquad (2.14)$$

for the conventional filter.

A similar expression can be derived for the DIC filter given by

$$\sigma_d^2 = \frac{G_1^2 G_2^2\,(1 + b) + \left[3\,G_2^2 + 2\,(3 - a - b)\right]\,(1 - a + b)}{6\,(1 - b)\,(1 - a + b)\,(1 + a + b)} \qquad (2.15)$$

where $G_1 = 2^{-b_1}$ and $G_2 = 2^{-b_2}$ are the gains of the integrator. Normally G_2 is of the order of 2^{-7} hence G_2^2 can be neglected in the above equation.

$$\therefore \sigma_d^2 \simeq \frac{3 - a - b}{3\,(1 - b)(1 + a + b)}$$

or

$$\sigma_d^2 \simeq \frac{2\,A_1 + A_2\,G_1}{3\,A_1\,(4 - 2\,A_1\,G_2 + A_2\,G_1\,G_2)} \qquad (2.16)$$

From this it can be seen that G_1 and G_2 affect the round off errors. The expressions for A_1 and A_2 are (from 2.12)

$$A_1 = (1 - b) / G_2 = (1 - b)\,2^{b_2} \qquad (2.17)$$

$$A_2 = (1 - a + b) / G_1\,G_2 = (1 - a + b)\,2^{b_1} \cdot 2^{b_2} \qquad (2.18)$$

Choice of b_2 :

From the above it can be seen that A_1 depends only on G_2 while A_2 depends upon both G_1 and G_2. Therefore depending upon the values assigned for G_1 and G_2 in the numerator of (2.16), the following situations will arise

a) $2\,A_1 \ll A_2\,G_1$ (2.19)

b) $2\,A_1 \gg A_2\,G_1$ (2.20)

and in the denominator, normally,

c) $2\,A_1\,G_2,\ A_2\,G_2\,G_1 \ll 4$ (2.21)

Using (2.19), (2.21) in (2.16)

$$\sigma_d^2 \simeq A_2 G_1 / 12 A_1 \qquad (2.22)$$

From this, small values of A_1 should be avoided, otherwise round off error increases. It was shown [4], from experimental studies, that values of $A_1 < 1$ should be avoided. Therefore $A_1 \geqslant 1$ is preferred to minimize the round off error. Using this condition in (2.17),

$$(1 - b)\, 2^{b_2} \geqslant 1$$

or

$$b_2 \geqslant \log_2 \left(\frac{1}{1 - b} \right) \qquad (2.23)$$

and the value of G_2 can be obtained by choosing a value for b_2 using the above equation.

Choice of b_1 :

A choice for b_1 can be obtained by setting $|y_1|_{max} \simeq |y_2|_{max}$ in Fig. 2.8 . Let $h_1'(n)$ and $h_2'(n)$ be the impulse responses from the input nodes to those at which $y_1(n)$ and $y_2(n)$ present [4].

Then

$$|y_1|_{max} \simeq |\Delta x|_{max} \sum_n |h_1'(n)| \qquad (2.24)$$

$$|y_2|_{max} \simeq |\Delta x|_{max} \sum_n h_2'(n) \qquad (2.25)$$

where

$$H_1'(Z) = \frac{(1 + c - d) + (ad - a - d - bc + 1) Z^{-1} + (1-a+b) Z^{-2}}{G_1 G_2 D(Z)}$$

$$= \sum_n h_1'(n) Z^{-n} \qquad (2.26)$$

$$H_2'(Z) = \frac{(c - 1) + (a - d) Z^{-1} + (1 - b) Z^{-2}}{G_2 D(Z)}$$

$$= \sum_n h_2'(n) Z^{-n} \qquad (2.27)$$

$$D(Z) = 1 - a Z^{-1} + b Z^{-2} \qquad (2.28)$$

Using these equations,

$$G_1 G_2 \sum_n |h_1'(n)| \simeq G_2 \sum_n |h_2'(n)|$$

or

$$2^{-b_1} \sum_n |h_1'(n)| \simeq \sum_n |h_2'(n)|$$

or

$$b_1 \simeq \log_2 \frac{\sum_n |h_1'(n)|}{\sum_n |h_2'(n)|} \qquad (2.29)$$

Choice for dynamic range scaling factor G_o:

From the Fig. (2.6), where the implementation of DIC filter using the input and output are in normal form is presented, the scaling factor G_o can be obtained by choosing $|\Delta y|_{max} = |\Delta x|_{max}$ or from the figure it is the same as

$$|Y_3|_{max} = |x|_{max} \qquad (2.30)$$

Using this G_o can be evaluated as

$$G_o \geqslant \frac{Y_{3\ max}}{x_{max}} \frac{1}{\sum_n |h(n)|} \qquad (2.31)$$

where h (n) is the impulse response of the filter. In this $\sum_n |h(n)|$ is the infinite summation and a reasonable value of n should be chosen such that the error in the summation is small. In the extreme case

$$G_o \simeq \frac{Y_{3\ max}}{x_{max}} \frac{1}{\sum_n |h(n)|} \qquad (2.32)$$

The value of G_o obtained from this equation can be adjusted to the nearest binary number in order to eliminate the multiplier. FORTRAN programs for calculation of these parameters are given in [4].

2.6 HIGH PASS FILTERING USING DIC SCHEME

The transfer function for a second order high pass filter is of the form

$$H(Z) = \frac{(1 - Z^{-1})^2}{1 - a Z^{-1} + b Z^{-2}} \qquad (2.33)$$

Comparing this with the transfer function of the second order filter

$$H(Z) = \frac{c - d Z^{-1} + Z^{-2}}{1 - a Z^{-1} + b Z^{-2}} \qquad (2.34)$$

we get c = 1; d = 2 ; using this the coefficients for DIC structure can be obtained as

$$B_o = 1; \quad B_1 = 0; \quad B_2 = 0$$
$$A_1 = (1 - b) / G_2; \quad A_2 = (1 - a + b) / G_1 G_2 \qquad (2.35)$$

The block diagram is shown in Fig. 2.12.

2.7 HIGH PASS FILTER FOR TRACK PROFILE MEASUREMENT

As described earlier, a one pole Butterworth high pass filter is first implemented for the real time application of track profile measurement. For this the high pass function considered will be

$$H(Z) = \frac{1 - Z^{-1}}{(w - 1) Z^{-1} + w + 1} \qquad (2.36)$$

For this filter, the working programs in 4 , are implemented on DEC-1090 computer and the following coefficients for the DIC scheme are obtained.

$$B_o = 1 \quad A_1 = 1 \quad b_2 = 5 \quad G_o = 1$$

and the DIC high pass filter will be as shown in Fig. 2.13.

The proposed system thus would consist of a first order Butterworth high pass filter using DIC structure and a simpson integrator. An assembly language program for 6800 microprocessor has been developed and is given in Appendix A. This program is for the composite filter consisting of a high pass filter and integrator. The programme is tested on 6800 microprocessor development system and the results and discussions are presented in Chapter (5).

After obtaining the different track irregularity parameters namely verticle profile of left and right

rails, through simple algebraic manipulation described in Chapter 1, one can derive all the other irregularities like unevenness, gauge etc.

In the following Chapter, the power spectral measurement of these irregularities through Fast Fourier Transform (FFT) technique is described.

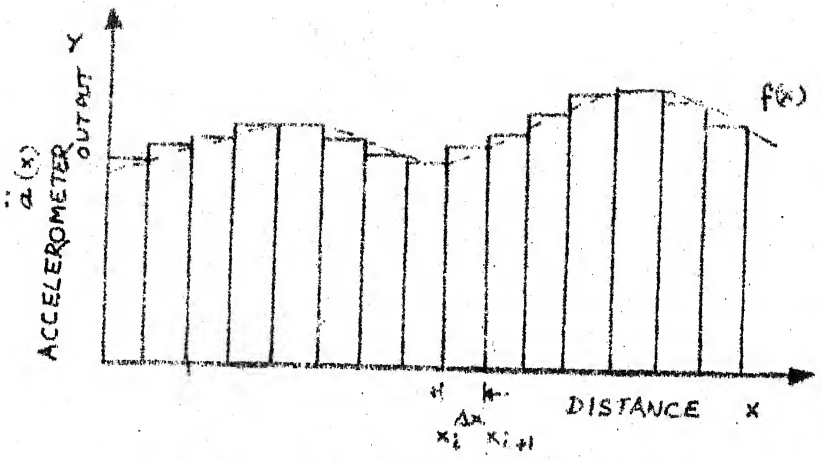


FIG. 2.1. SIMPSON INTEGRATION

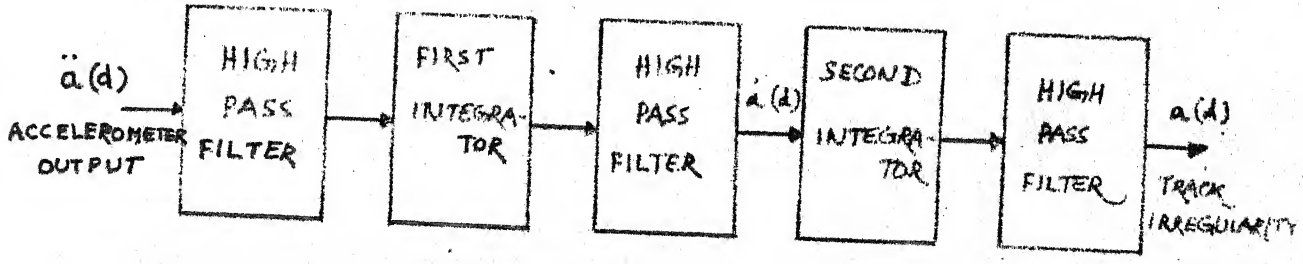


FIG. 2.2. FILTERING SCHEME

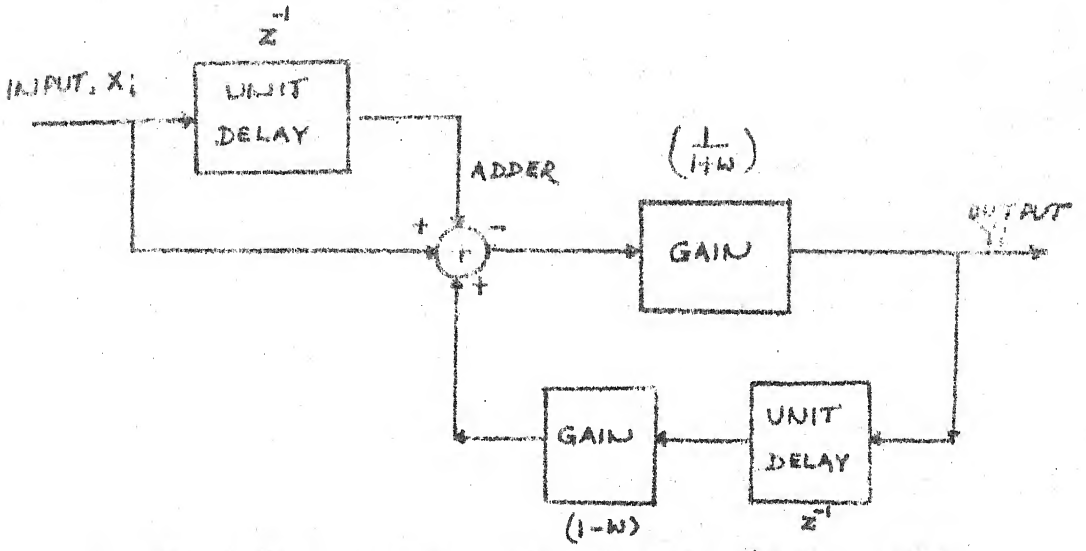


FIG. 2.3. REALIZATION OF SINGLE POLE HIGH PASS BUTTERWORTH DIGITAL FILTER.

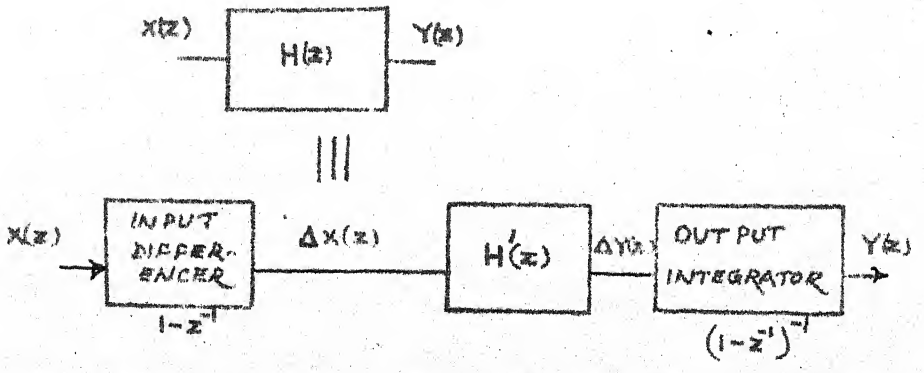


FIG. 2.4. ILLUSTRATION OF THE EQUIVALENCE BETWEEN CONVENTIONAL AND DIC FILTERS.

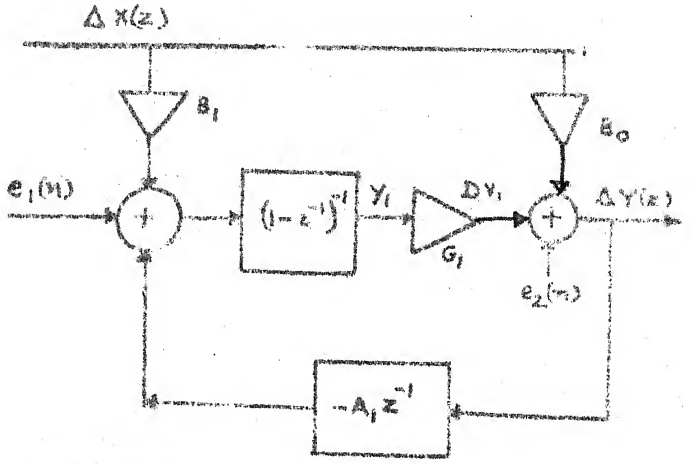


FIG. 2.5 BLOCK DIAGRAM FOR THE DIS STRUCTURE OF FIRST ORDER FILTER.

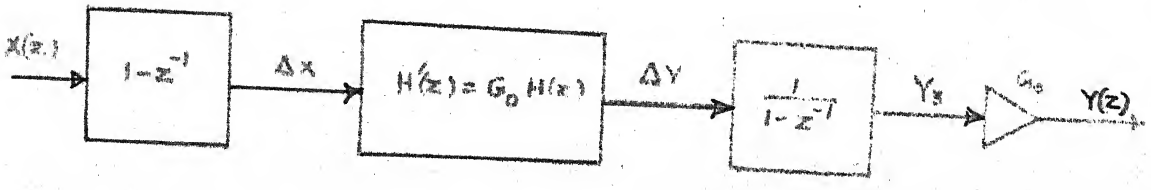


FIG. 2.6. BLOCK DIAGRAM OF DIC FILTER WITH INPUT AND OUTPUT IN NORMAL FORM.

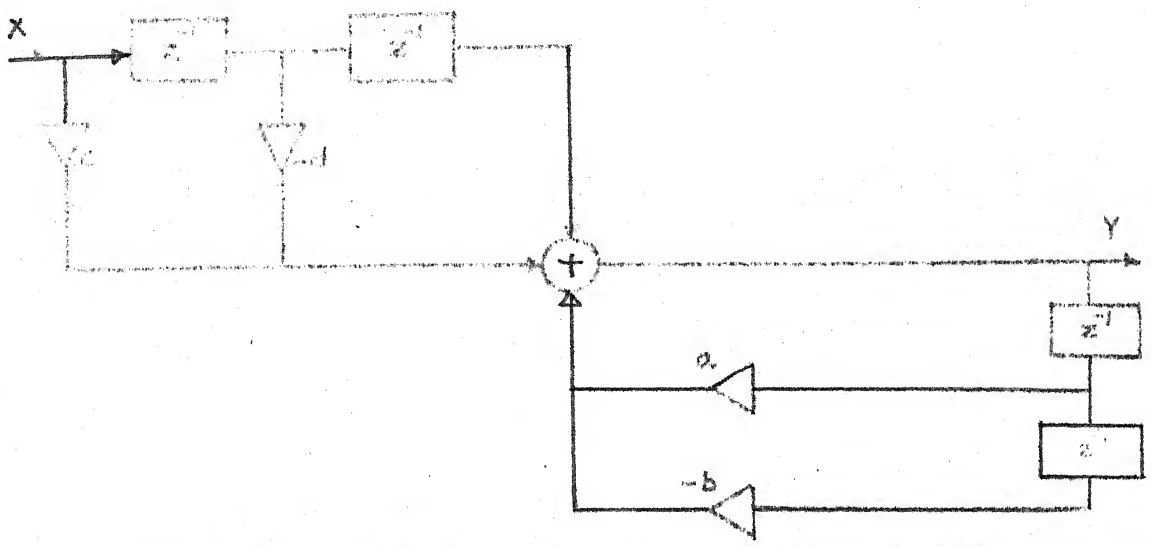


FIG. 2.7. SECOND ORDER CONVENTIONAL FILTER.

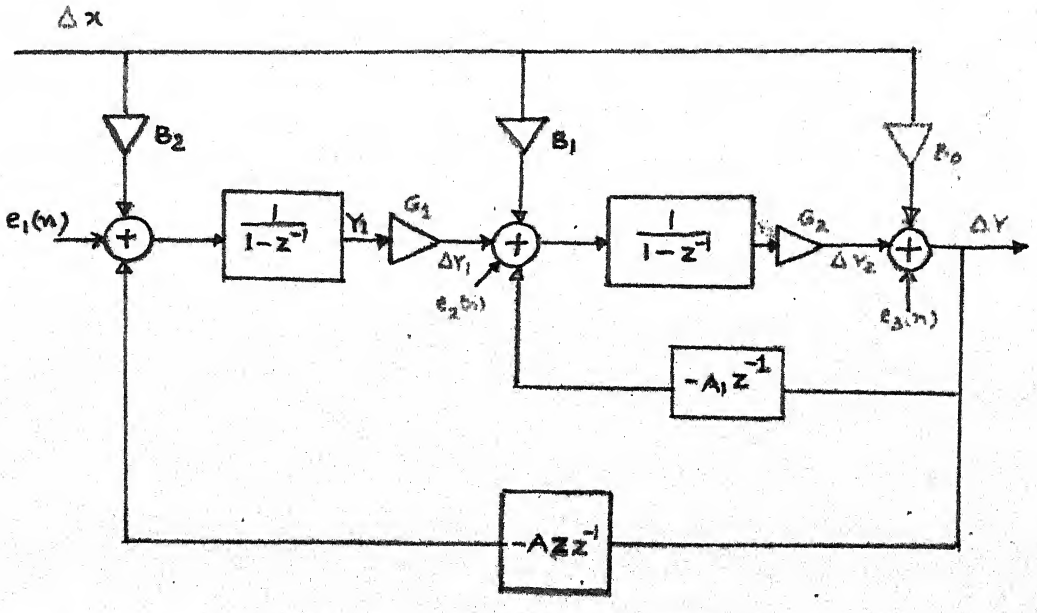


FIG. 2.8. SECOND ORDER DIC FILTER.

$$Y_{q'} = (X_q \cdot a)_{q'}$$

FIG. 2.9. FIXED POINT QUANTIZATION NOISE MODEL.

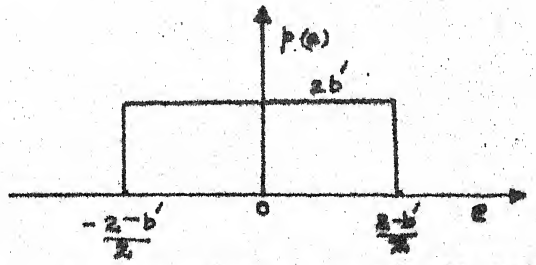


FIG. 2.10. QUANTIZATION NOISE PROBABILITY DENSITY FUNCTION FOR THE ROUNDING OPERATION.

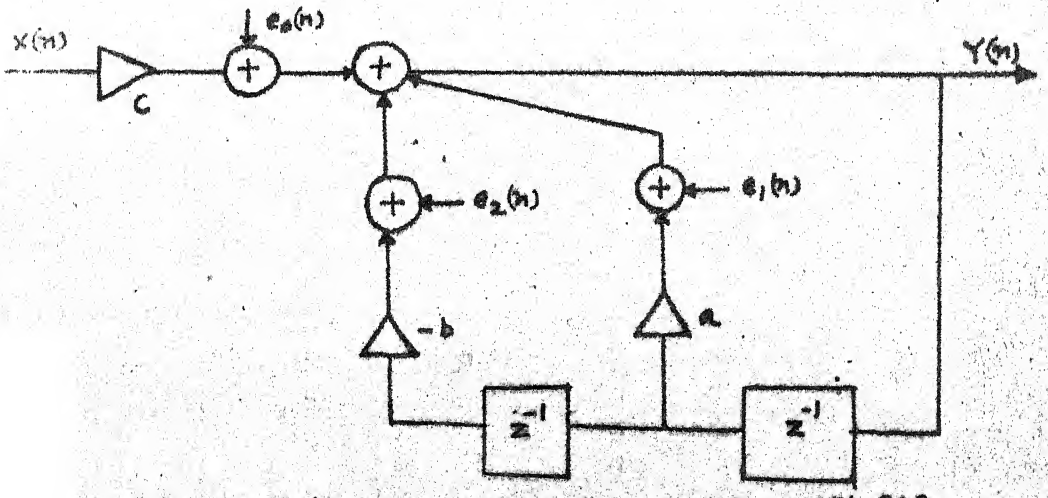


FIG. 2.11. QUANTIZATION NOISE MODEL FOR SECOND ORDER FILTER $H(z) = \frac{c}{1 - a z^{-1} + b z^{-2}}$.

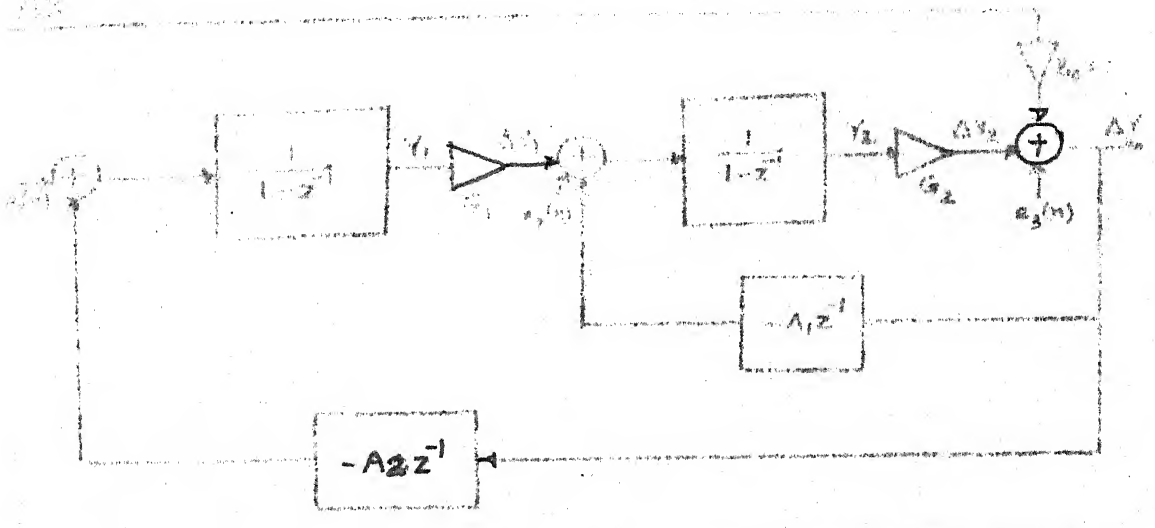


FIG. 2.12 BLOCK DIAGRAM OF A SECOND ORDER DIC FILTER.

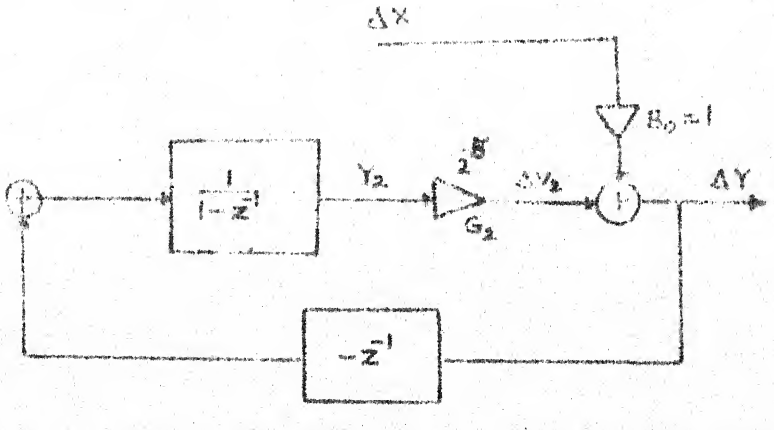


FIG. 2.13. FIRST ORDER HIGH PASS BUTTERWORTH FILTER - DIC STRUCTURE.

CHAPTER 3

POWER SPECTRAL MEASUREMENT OF TRACK IRREGULARITIES

3.1 INTRODUCTION

As the trains move on irregular tracks the comfort of the passenger is affected. Standards have been established for minimum comfort to be provided. Equivalently standards are also laid down for categorization of rail tracks through roughness factors. These factors are intimately connected to the Power Spectral Density (PSD) of the random process correspondent to track irregularities.

In Chapter 2 the methods of obtaining the waveforms corresponding to the track irregularities are discussed. Using this the power spectral density associated with the irregularity is to be obtained. A set of N samples associated with the track irregularity is available to estimate the power spectral density. In the following sections, the estimation of PSD through the use of Fast Fourier Transform (FFT) is discussed. The implementation of FFT on 6800 microprocessor is also described.

3.2 POWER SPECTRAL DENSITY (PSD)

The random process is said to be 'strongly stationary' when all the possible moments of the process

are time invariant. The process is said to be 'weakly stationary' in wide sense if the first two moments namely mean and autocorrelation of the process are time invariant. Practically it was observed that track irregularity parameters constitute a weakly stationary random process.

The PSD of a process is defined by

$$S_X(\omega) = \int_{-\infty}^{\infty} R_X(\tau)\, e^{-j\omega\tau}\, d\tau \qquad (3.1)$$

where X is the random process, $S_X(\omega)$ is the PSD and $R_X(\tau)$ is the autocorrelation of X. The PSD is non negative for all ω and even function. Because of this the PSD is defined only over the positive frequencies from Zero to $+\infty$ and is referred to as 'one sided spectrum'. The ordinates of one sided spectrum will be double the right half of mathematically generated two sided spectral density, and is used here. The important constraint underlying the definition of PSD is that the sample function (or realization) of the process should be a record from a stationary random process of infinity. However, the concept of PSD is also equally applicable to weak stationary process. Since it is not possible to take a record length of infinity, a realizable length L will be chosen to make the definition of PSD physically realizable and to obtain a consistent estimate of PSD.

3.2.1 Mathematical Analysis [5]

Although the concept of random process has been developed for continuous-time random signals, the same can be easily applied to the case of discrete-time random signals. The mean and autocorrelation of a discrete process X (nT) can be expressed as

$$E\left[X\,(nT)\right] = \int_{-\infty}^{\infty} x\; p\,(x;\, nT)\; dx = \eta_x \qquad (3.2)$$

$$R_X\,(kT) = E\left[X\,(nT)\; X\,(nT + kT)\right] \qquad (3.3)$$

or $$R_X\,(k) = E\left[X_n\, X_{n+k}\right] \qquad (3.4)$$

where k is the time separation or lag.

In the discrete case, the N samples from a realization of a zero-mean stationary random process will be used to 'estimate' its autocorrelation function, which is defined as

$$R\,(k) = E\left[x_n\, x_{n+k}\right] \qquad (3.5)$$

First the estimation $R\,(0) = \sigma^2$ will be considered. Let the estimate be

$$\hat{\sigma}^2 = R_N\,(0) = \frac{1}{N} \sum_{0}^{N-1} x_i^2 = \frac{x_o^2 + x_1^2 \cdots + x_{N-1}^2}{N} \qquad (3.6)$$

In this, if k is large x_i and x_{i+k} will be effectively independent eventhough the realizations x_o^2, x_1^2, ... need not be independent.

The estimate $\hat{\sigma}^2$ is eq. (3.6) is unbiased (for large N) since $E[\hat{\sigma}^2] = \sigma^2$ and also consistent since $\lim_{N\to\infty} E\left[(\hat{\sigma}^2 - \sigma^2)^2\right] = 0$ i.e. variance is zero. Hence for large N, the $\hat{\sigma}^2$ from a single realization has a high probability of nearing the true value σ^2.

In the similar lines, the estimate for $R_N(k)$, the autocorrelation function at other lags can be proposed as

$$R_N(k) = \frac{x_0 x_{|k|} + x_1 x_{|k|+1} + \cdots + x_{N-1-|k|} x_{N-1}}{N} \tag{3.7}$$

$$= \frac{1}{N} \sum_{i=0}^{N-|k|-1} x_i x_{i+|k|} \quad k = 0, \pm 1, \pm 2, \cdots \pm N-1 \tag{3.8}$$

which averages all possible products of samples separated by a lag of k and for large k there are very few possible products. For $k \geqslant N$, there are no possible pairs with this lag available, hence it will be assumed that the estimate of $R_N(k)$ is zero.

$$\therefore R_N(k) = \begin{cases} \frac{1}{N} \sum_{i=0}^{N-|k|-1} x_i x_{i+|k|} & k = 0, \pm 1, \pm 2 \cdots \pm N-1 \\ 0 & k \geqslant N \end{cases} \tag{3.9}$$

$R_N(k)$ is an evenfunction of k, as is true in the case of $R(k)$. Here N denotes that the realization is of N samples in length.

The mean value of the above estimate is

$$E\left[R_N(k)\right] = \frac{1}{N}\sum_{j=0}^{N-1-|k|} E\left[x_j\, x_{j+|k|}\right]$$

$$= \left(1 - \frac{|k|}{N}\right) R(k) \qquad (3.10)$$

Thus the sample autocorrelation function in (3.9) is biased since the mean of the estimate is not the true autocorrelation function R (k) at lag k. However, as $N \to \infty$, the term $|k|/N$ vanishes. Hence $R_N(k)$ is assymptotically unbiased. An unbiased estimate can be easily obtained by dividing the sum in (3.10) by $(N - |k|)$ rather than N. Then

$$E\left[R_N(k)\right] = \frac{1}{N-|k|}\sum_{j=0}^{N-1-|k|} E\left[x_j\, x_{j+|k|}\right] \qquad (3.11)$$

However it was shown [5] that for the estimate of (3.10) variance is small for $N \gg k$ ior which bias is also small. Therefore, unless a large number of lag products is available for an estimate at a k, the resulting estimate $R_N(k)$ will not be reliable.

3.2.2 Estimates of PSD

The power spectral density and autocorrelation function of a realization are related through the Discrete Fourier Transform (DFT), given by

$$S(\omega) = \sum_{k=-\infty}^{\infty} R(k)\, e^{-jk\omega T_s} \qquad (3.12)$$

Considering the estimate of $R(k)$ as $R_N(k)$, given by (3.9), the spectral estimate $S_N(\omega)$ will be,

$$S_N(\omega) = \sum_{k=-\infty}^{\infty} R_N(k)\, e^{-jk\omega T_s} \tag{3.13}$$

The spectral estimate $S_N(\omega)$ is also related to the data samples $X_N(\omega)$ through the DFT,

$$S_N(\omega) = \frac{1}{N}\left| \sum_{n=0}^{N-1} x_n\, e^{-j\omega T_s}\right|^2 = \frac{1}{N}\left|X_N(\omega)\right|^2 \tag{3.14}$$

This can be proved as follows. Consider the truncated signal

$$x_n^N = \begin{cases} x_n & n = 0, 1, \ldots N-1 \\ 0 & \text{otherwise} \end{cases} \tag{3.15}$$

Then

$$R_N(k) = \frac{1}{N} \sum_{n=-\infty}^{\infty} x_n^N\, x_{n+k}^N \tag{3.16}$$

From (3.13) $S_N(\omega)$ is given by

$$S_N(\omega) = \frac{1}{N} \sum_{k=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} x_n^N\, x_{n+k}^N\, e^{-jk\omega T_s} \tag{3.17}$$

$$= \frac{1}{N} \sum_{n=-\infty}^{\infty} x_n^N\, e^{j\omega nT_s} \sum_{k=-\infty}^{\infty} x_{n+k}^N\, e^{-j\omega(n+k)T_s} \tag{3.18}$$

$$= \frac{1}{N} X_N(\omega)\, X_N^*(\omega) = \frac{1}{N}\left|X_N(\omega)\right|^2 \tag{3.19}$$

in which $X_N(\omega) = \sum_{n=-\infty}^{\infty} x_n^N\, e^{-jn\omega T_s} = \sum_{n=0}^{N-1} x_n\, e^{-jn\omega T_s}$

Considering the estimate $S_N(\omega)$, the mean of $S_N(\omega)$ is,

$$E[S_N(\omega)] = \sum_{k=-\infty}^{\infty} E[R_N(k)]\, e^{-jk\omega T_s} \qquad (3.20)$$

From (3.10),

$$E[S_N(\omega)] = \sum_{k=-N}^{N} R(k)\left(1 - \frac{|k|}{N}\right) e^{-jk\omega T_s} \qquad (3.21)$$

Thus the spectral estimate is the biased estimate of $S(\omega)$, which differs from $S(\omega)$ by the term

$$v_k^N = \begin{cases} \left(1 - \frac{|k|}{N}\right) & |k| \leq N \\ 0 & |k| > N \end{cases} \qquad (3.22)$$

However the $S_N(\omega)$ is assymptotically unbiased since the term $|k| / N$ vanishes for large N. The term v_k^N is called the 'window function'. The presence of this term in spectral estimate causes the power that presents in a particular frequency range to be smeared out over a wider range. This spreading of the power is also called as 'LEAKAGE'. For having an accurate estimate of PSD, suppression of sidelobes or reduction of leakage is essential. This can be achieved by using spectral windows on raw power spectral density. There are many window functions available [5] and the following are the most popular windows that are practically used.

a) Rectangular

b) Hann

c) Hamming

d) Blackman

e) Kaiser

Out of which Hann and Hamming windows are mostly used in many applications.

3.3 COMPUTATION OF PSD

There are three methods available to evaluate the PSD

a) To compute the autocorrelation function and then finding the Fourier transform.

b) Direct spectra using Fast Fourier Transform Technique.

c) Filtering the signal through a large number of band filters and dividing the output through each band by the Bandwidth.

Out of these the second method which finds the PSD using Fast Fourier Transform Technique is found to be the fastest. Hence the same will be used in this application. For smoothing out the raw power spectral density, Hann window function will be used as it is easier to implement on a micro computer. The Hann window function in the frequency domain is given by

$$U_T(f) = \frac{1}{2}\hat{U}_T(f) + \frac{1}{4}\hat{U}_T\left(f - \frac{1}{2T}\right) + \frac{1}{4}\hat{U}_T\left(f + \frac{1}{2T}\right) \quad (3.23)$$

where $U_T(f)$ is the modified function

$\hat{U}_T$ (f) is the raw function

2T is the record length (from -T to T)

The computation procedure for PSD is then will be as follows:

a) Obtain a data sequence of N samples.

b) Detrend the data using appropriate filter (removal of Mean)

c) Compute the Discrete Fourier Transform of the sequence and find out the real and imaginary Fourier coefficients.

$$\mathcal{F}(k) = \sum_{i=0}^{N-1} X_i \, e^{-j2\pi ki/N} = A(k) + j\,B(k)$$

d) Compute the one sided raw power spectral density using the following equation

$$S_X(k) = \frac{2}{N}\left[A^2(k) + B^2(k)\right]$$

e) Using appropriate spectral window on raw estimates, compute the smoothened Power Spectral Density.

f) Calculate smoothened PSD for each realization of the ensemble and take an average to get the PSD of track irregularity.

3.4 FAST FOURIER TRANSFORM (FFT) Algorithm [1], [2]

The power spectral measurement involves the computation of Discrete Fourier Transform of the data sequence. The direct evaluation of the DFT involves N

complex multiplications and (N - 1) complex additions for each value of X (k) and since there are N such values to be determined, N^2 multiplications and N (N - 1) additions are necessary. For larger N, direct evaluation of DFT involves considerable amount of computation . Fast Fourier Transform (FFT) algorithm evaluates the DFT of a sequence in a efficient way by reducing the total number of computations. The FFT algorithm does this by splitting the N point sequence into two N/2 point sequences.

Let the N point sequence be $x(n)$ $n = 0, 1, \ldots N-1$ where N is a power of two. Defining the two N/2 point sequences $x_1(n)$, $x_2(n)$ as the even and odd members of x (n) respectively

$$\begin{aligned} x_1(n) &= x(2n) \qquad n = 0, 1, \ldots \frac{N}{2} - 1 \\ x_2(n) &= x(2n+1) \qquad n = 0, 1, \ldots \frac{N}{2} - 1 \end{aligned} \tag{3.24}$$

Then the N - point DFT of $\{x(n)\}$ can be expressed as [1],

$$X(k) = X_1(k) + W_N^k X_2(k) \tag{3.25}$$

where $X_1(k)$ and $X_2(k)$ are the N/2 point DFT's of $x_1(n)$ and $x_2(n)$ respectively. Since X (k) is defined for $0 \leq k \leq N - 1$ and $X_1(k)$, $X_2(k)$ are defined for $0 \leq k \leq \frac{N}{2} - 1$. To interpret for the values of $k \geq \frac{N}{2}$, the following equation can be used

$$X(k) = \begin{cases} X_1(k) + W_N^k X_2(k) & 0 \leq k \leq \frac{N}{2} - 1 \\ X_1(k - \frac{N}{2}) + W_N^k X_2(k - \frac{N}{2}) & \frac{N}{2} \leq k \leq N - 1 \end{cases} \tag{3.26}$$

This can be obtained using the periodicity property of DFT i.e. $W_N^{k+N/2} = -W_N^k$. In a similar manner the N/2 point DFT can be expressed as a combination of N/4 point DFT, i.e.,

$$\begin{aligned} X_1(k) &= A(k) + W_{N/2}^k B(k) \\ X_2(k) &= A(k) + W_N^{2k} B(k) \end{aligned} \tag{3.27}$$

where $A(k)$ and $B(k)$ are the N/4 point DFT's of the even and odd members of $x_1(n)$ respectively. The above process can be continued until a two point DFT arrives. A two point DFT, $F(k)$, $k = 0, 1$ may be evaluated using no multiplications as

$$\begin{aligned} F(0) &= f(0) + f(1) W_8^0 \\ F(1) &= f(0) + f(1) W_8^4 \end{aligned} \tag{3.28}$$

where $f(n)$, $n = 0, 1$ is the two point transform being transformed.

Since $W_8^0 = 1$ and $W_8^4 = -1$ there are no multiplications involved in computing the above equation. The total number of operations using this scheme is of the order of $N \log_2 N$. Thus it can be seen that when N is large, there is a great reduction in terms of computation.

The above described algorithm is also called as Decimation-in-Time (DIT) algorithm. Since at each stage of the process the input sequence is splitted into

smaller sequences for processing. Another popular form of the FFT algorithm is Decimation-in-Frequency (DIF) algorithm. For this version, the input sequence $x(n)$ is partitioned into two sequences each of length N/2. The first sequence $x_1(n)$ consists of the first two N/2 points of $x(n)$ where as the second sequence $x_2(n)$ consists of the last N/2 points of $x(n)$. The same mathematical analysis can also be done for DIF algorithm [1]. The diagram in Fig. 3.1 shows the DIT and DIF algorithms for N = 8.

From these illustrations, it can be observed that, for DIT scheme, for the output to be in natural order the input should be shuffled where as reverse is the case for DIF scheme. Hence whether it is a DIT or DIF shuffling of the data is necessary. Although shuffled, the order of the shuffled sequence can be determined in a simple manner. When N is a power of 2, the shuffled sequence will be the bit reversed number of the index when it is represented in Binary. For example the following Table shows the shuffled sequence for N = 8.

Table 3.1

Index	Binary Representation	Bit Reversed Binary	Bit Reversed Index
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

The same procedure can be adopted to any index when N is a power of 2. To calculate the FFT coefficients $\{W_N^k\}$, either the coefficients can be formed as a table and then be referenced at various stages during the computation or these coefficients can be computed using

$$W_N^k = \text{Cos}\left[\left(\frac{2\pi}{N}\right)k\right] - j\,\text{Sin}\left[\left(\frac{2\pi}{N}\right)k\right]$$

where in the first method the extra storage is needed while in the second case computation time is involved.

The FFT algorithms can be further categorized into

a) Inplace Algorithm

b) Natural Input-Output Algorithm

a) Inplace Algorithm: In place Algorithm is the one in which a given component of any intermediate vector may be stored in the same location occupied by the corresponding component of the preceding vector. This requires less storage but computational time is higher than the Natural Input Output algorithm. Other characterstic is that either the output spectrum appears in an unnatural order or that the input data should be arranged before entering the computational array, as mentioned earlier. This shuffling process is also called as scrambling operation.

b) Natural Input-Output Algorithm: This is the one in which a given component of any intermediate vector may not be

stored in the same location occupied by the corresponding component of the preceding vector, thus requiring the extra memory for storing the Intermediate Results. An N-point FFT (N real and N imaginary) will require 4N words of memory as compared to 2 N words required for Inplace algorithm. These algorithms maintain a natural input-output order and do not require scrambling operation and as such are faster when compared to the inplace algorithm.

Considering the present application, inplace algorithm is adopted to reduce the memory constraints of the system.

Basic Properties of Inplace Algorithm:

From the signal flowgraph the following points can be observed.

a) There are $r = \log_2 N$ number of computational arrays.

b) In each array every node has two incoming paths and two outgoing paths.

c) In each array, there are two input nodes whose input paths originate from the same pair of nodes in the previous array. Two such nodes are grouped as 'a dual node pair'. In any array there are N/2 pairs of dual nodes.

d) Each array requires N/2 complex multiplications and N complex additions. Therefore the total number of multiplications and additions required are $(N/2) \log_2 N$ and $N \log_2 N$ respectively.

e) Computation of dual node pair requires only one multiplication and two additions. If the weighting factor at one of the nodes in a dual pair is W^p, then the weighting factor at the other node of the pair is $W^{p+N/2}$. Then

$$X_l (k) = X_{l-1} (k) + W^p X_{l-1} (k + N/2^l)$$

$$X_l (k + N/2^l) = X_{l-1} (k) - W^p X_{l-1} (k + N/2^l) \quad (3.29)$$

$$\because W^{p+N/2} = - W^p$$

where $X_l (k)$ indicates k component in the l^{th} array.

f) The spacing between dual node pair differs from array to array. In the l^{th} array ($l = 1, 2, \ldots r$) the spacing is $N/2^l$ means $X (k)$ and $X (k + N/2^l)$ constitute a dual node pair.

g) To evaluate the value of p which is the exponent of W for any index in a given array, represent k, the node index in the l^{th} array in binary form with r bits, retain the most significant l bits and add (r - l) leading zeros to form a r bit binary number. Reverse the bit order of the resulting number and the decimal equivalent of

the final binary number gives the index p. The weighting factor for the k^{th} node of the l^{th} array is W^p.

h) The output after r arrays is in scrambled form. To unscramble the output X (k) represent the index k in binary form with r bits and reverse the bit order. The resulting decimal number is the index n of X (n).

3.4 IMPLEMENTATION OF FFT ON 6800 MICROPROCESSOR

As it was found [3], considering the accuracy point of view, a 16 bit word length should be chosen to represent each data word. For a real time application, though less accurate, fixed point arithmetic will be faster than floating point arithmetic. Hence fixed point arithmetic is chosen for this application where the data is represented in Two's complement form, and the binary point is assumed to lie to the left of leftmost magnitude bit. As the execution of the program moves from stage to stage, the magnitudes of the numbers in the sequence generally increase hence there is a possibility of overflow during different stages of computation. To prevent this appropriate techniques of scaling should be adopted in fixed point arithmetic.

Scaling [3] : The power of 2 algorithm, as mentioned earlier, operates on two complex numbers. It takes

these two numbers and produces two new complex numbers which replace the original ones in the sequence.

Let $X_m(i)$ and $X_m(j)$ be the original complex number. The new complex pair is given by

$$X_{m+1}(i) = X_m(i) + W X_m(j)$$
$$X_{m+1}(j) = X_m(i) - W X_m(j) \qquad (3.30)$$

The algorithm goes through the entire sequence of N numbers, taking two at a time. When $N = 2^M$, total number of computational stages is M. With the assumption of binary point lying towards extreme left, the relation between the numbers in m and m + 1 stage is as shown in Fig. 3.2. The outside square gives the possible values of $\mathrm{Re}[X_m(i)] < 1$ and $\mathrm{Im}[X_m(i)] < 1$ and the circle inscribed in the square gives the region $X_m(i)$ 1. Similarly the inside circle gives the region $|X_m(i)| < 1/2$. If the numbers $X_m(i)$, $X_m(j)$ lie in the region $|X_m(i)| < 1/2$, then $X_{m+1}(i)$, $X_{m+1}(j)$ will be in the larger circle hence overflow condition doesn't arise. However, if $X_m(i)$ and $X_m(j)$ are inside the smaller square, it is possible that $X_{m+1}(i)$, $X_{m+1}(j)$ to be outside the larger square hence overflow arises. So the input sequence should be controlled such that $|X_m(i)| < 1/2$ to prevent overflow . There are three techniques of scaling, which can be used for this, namely

a) Shifting right one bit at every iteration

b) Controlling the sequence so that $|X_m(i)| < 1/2$

c) Testing for an overflow.

Out of these, the first method is the simplest and easy to adopt for microprocessor applications and hence it will be used in this implementation.

A flow chart, and the corresponding loops in the FFT in place program is given in Fig. 3.3. As stated earlier in the FFT routine, the k component in the l array is given by

$$X_l(k) = X_{l-1}(k) + W^p X_{l-1}(k + N/2^l)$$
$$X_l(k + N/2^l) = X_{l-1}(k) - W^p X_{l-1}(k + N/2^l) \qquad (3.31)$$

where W^p is the multiplier and is a complex number, given by

$$W^p = e^{\frac{-j\,2\pi}{N} p} = \cos\left(\frac{2\pi}{N}\right) p - j \sin\left(\frac{2\pi}{N}\right) p$$

As it can be seen, for generating the weights of W, sine and cosine values are required, and since calculation of sine and cosine functions is time consuming, these values can be generated and stored in the system memory before entering into the FFT program, for reducing the execution time. For this N/2 values of SINE and N/2 values of COSINE are required from 0 to $\frac{2\pi}{N} \times \left(\frac{N}{2} - 1\right)$ in a stepsize of $\frac{2\pi}{N}$, which corresponds to N/2 weights of W from W^0, W^1, ... to $W^{N/2-1}$. This requires extra N words for storing the values of SINE and COSINE.

However, values of sine and cosine can be derived by storing only N/4+1 values of sine from 0 to $\pi/2$ in a stepsize of $2\pi/N$ i.e. from W^0, W^1, ... to $W^{N/4}$, as described below. For N = 1024, this needs the values of sine to be stored from 0 to $\pi/2$ in a stepsize of $2\pi/1024$ which corresponds to W^0, W^1, ..., to W^{256}.

Calculation of Sine Values (N = 1024)

a) First whether the index ≤ 256 or > 256 is to be checked.

b) If index is ≤ 256, then use the index as it is to fetch the appropriate sine value from the array.

c) If index is > 256, the index is subtracted from 514 (in general from 2 (N/4+1)) and this value is used as index to fetch the appropriate sine value.

Calculation of Cosine Value (N = 1024)

a) The program should check whether index ≤ 256 or > 256 (in general index $\frac{N}{4}$ or $\frac{N}{4}$).

b) If index is ≤ 256, subtract the index from 256 (generally from N/4) and this value will be used as index to fetch the value from Sine array.

c) If the index is > 256, subtract 256 (generally N/4) from the index and this value will be used as the index to fetch the value from Sine array which gives appropriate Cosine values.

By this method, the memory required will be N/4+1 words for storing Sine values to generate the weights which will cut down the memory from N words to N/4+1 words.

Organization of the FFT Program

The FFT subroutine assembly language program mainly calls the following subroutines:

a) COMPLEMENT ROUTINE (COMPL)

This routine finds the Two's complement of 16 bit number when called.

b) OVERFLOW ROUTINE (OFL)

This routine shifts the 16 bit number to right by one bit, equivalently it divides the number by 2.

c) POWER OF TWO ROUTINE (LOG2N)

This finds the $\log_2 N$ where N is the number of samples and is a power of Two.

d) DOUBLE PRECISION MULTIPLY ROUTINE (DPMUL)

This multiplies the two 16 bit numbers and produces a 16 bit product using Booth's algorithm. Negative numbers are represented in Two's complement form.

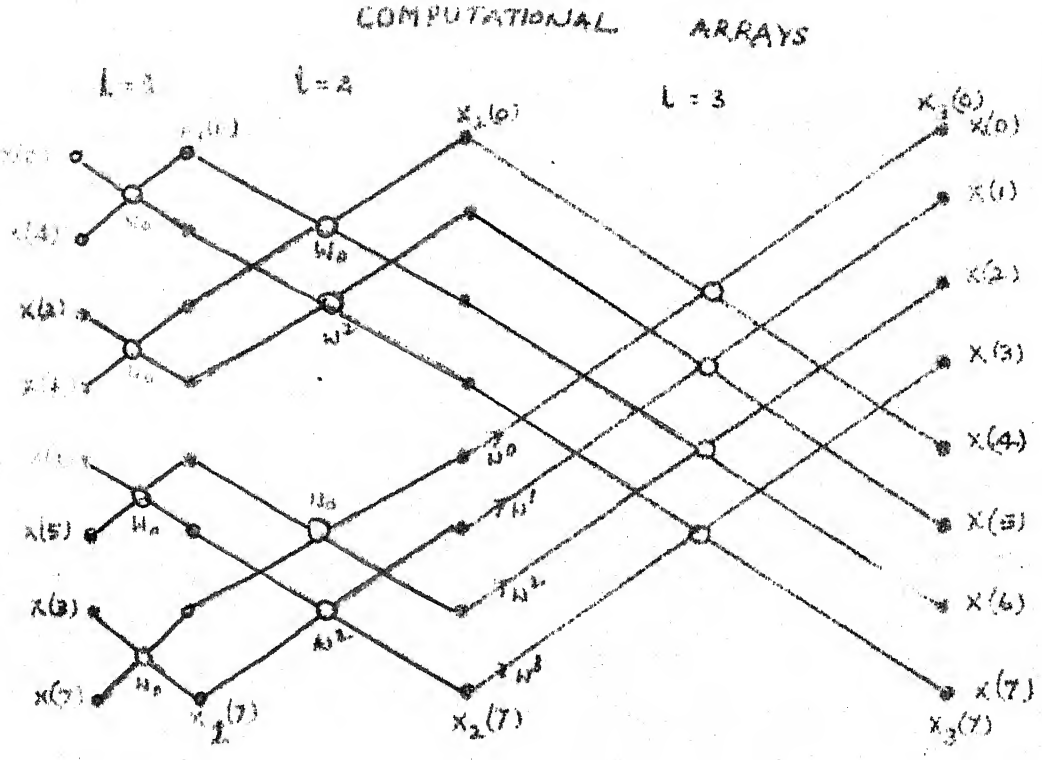
e) BIT REVERSAL ROUTINE (BITR)

This finds the Bit reversed number of a 16 bit number.

f) UNSCRAMBLING ROUTINE (UNSCM)

For inplace algorithm, since the output is in scrambled form, this routine orders the output sequence.

A complete assembly listing of FFT for 6800 microprocessor applications which can be used upto N = 1024 is given at appendix. The listing to find the power spectral density which calls the FFT routine is also included in the Appendix. The memory requirements and the results are presented, details are discussed in Chapter 5.



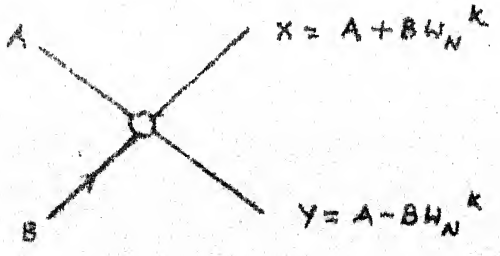
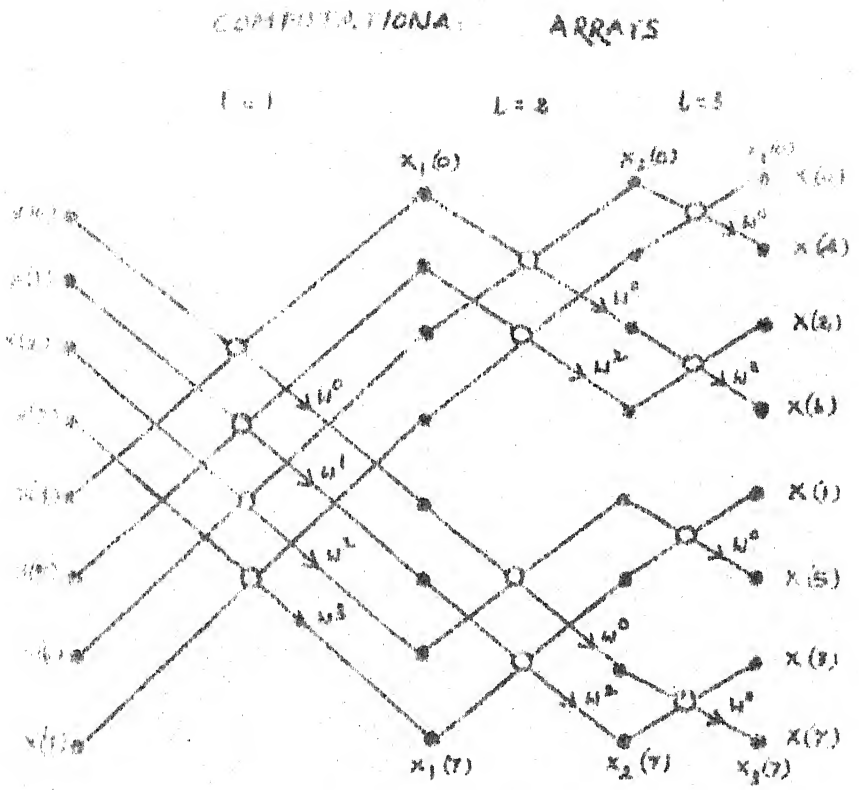


FIG. 3.1 (a) BUTTERFLY FOR DIT:
COMPLETE 8 POINT DECIMATION-IN-TIME FFT



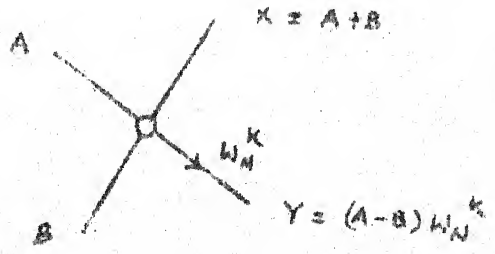


FIG. 3.1 (b) BUTTERFLY FOR DIF

COMPLETE 8 POINT DECIMATION-IN-FREQUENCY FFT.

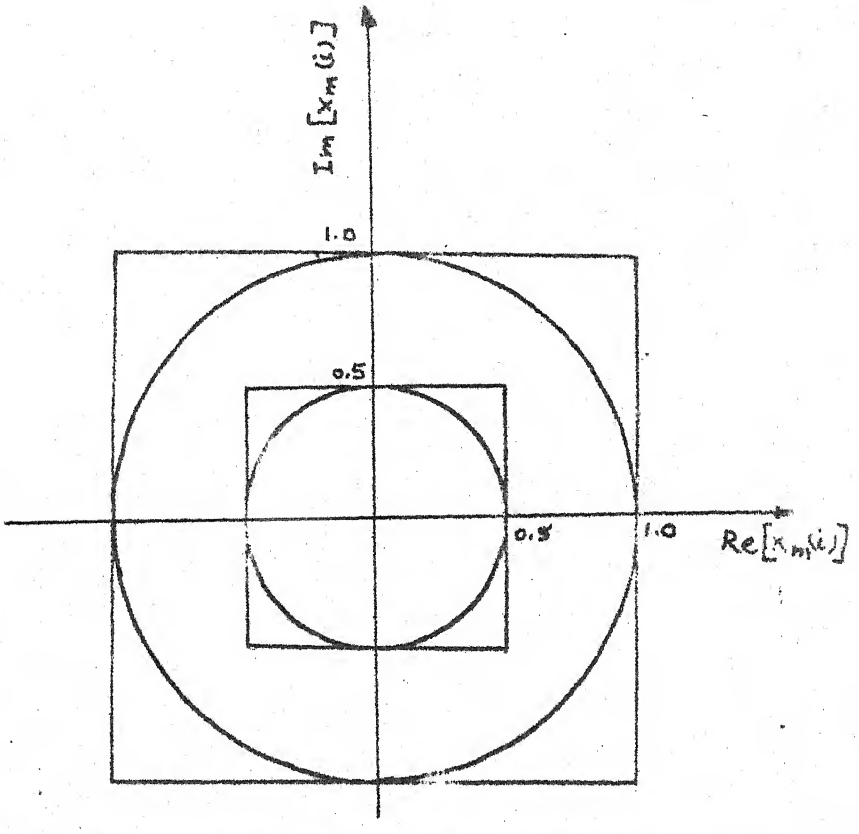


FIG. 3.2. RELATIONSHIP BETWEEN THE NUMBERS IN M AND M+1 STAGE.

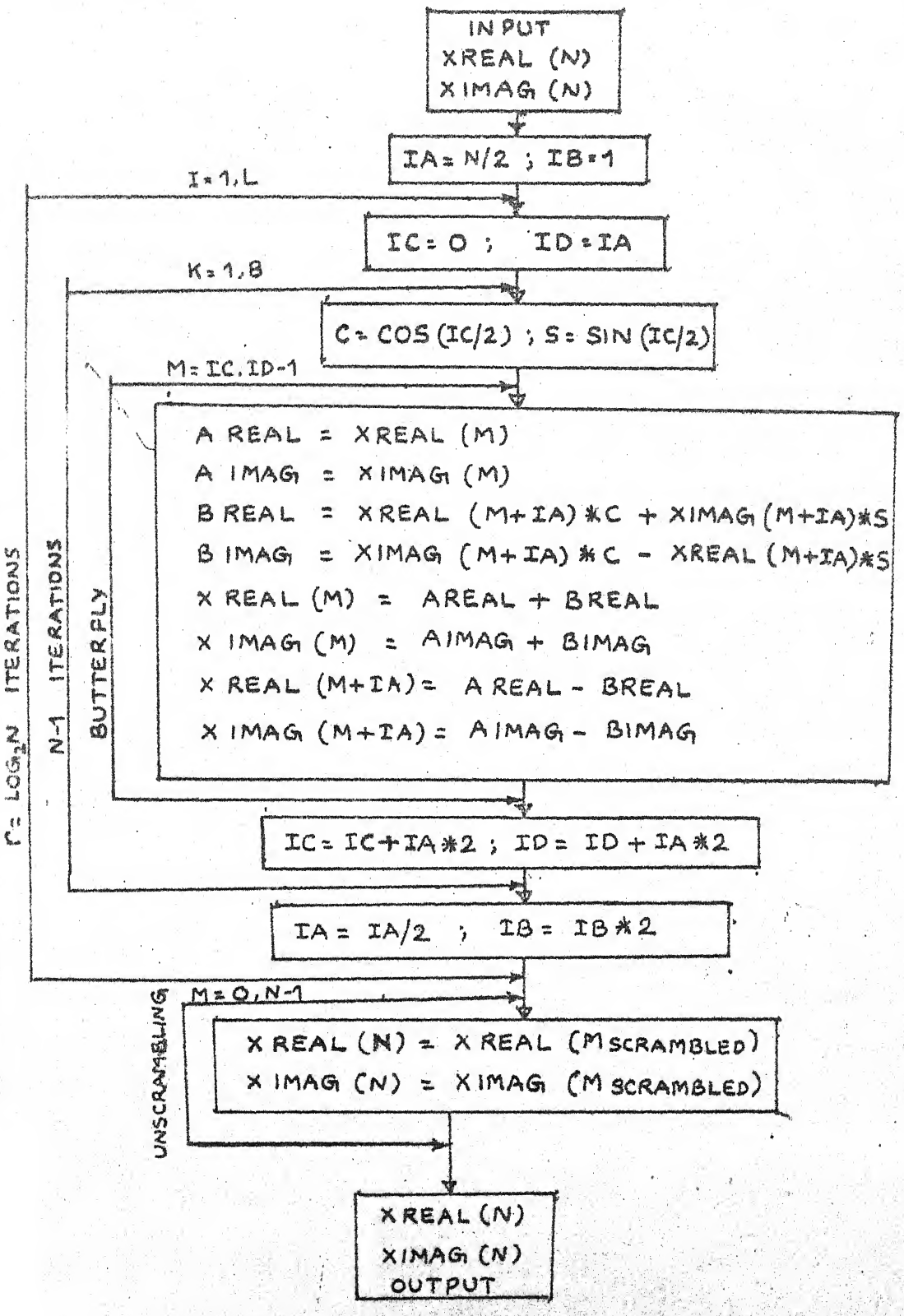


FIG. 3.3 FFT INPLACE ALGORITHM FLOWCHART.

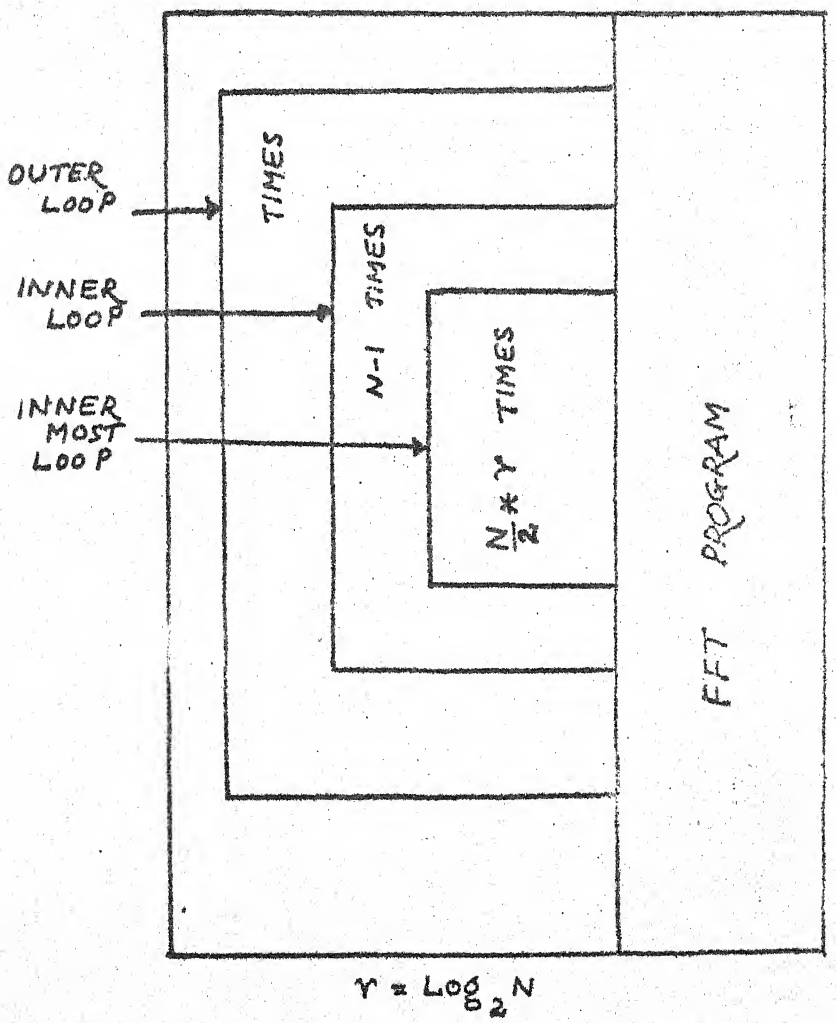


FIG. 3.4 FFT PROGRAM AND ITS LOOPS.

CHAPTER 4

MICROCOMPUTER BASED SYSTEM

4.1 INTRODUCTION

The technological revolution, spearheaded by the MICROPROCESSOR, a single chip central processing unit has rendered cost effective real time signal processing practical. A typical real time signal processing scheme like the track profiles measurement using large main frame computers or minicomputers would have been an expensive proposition and therefore impractical for routine track evaluation. With today's microprocessors, a low cost, completely independent on line dedicated signal processing system can be built for such applications.

The 6800 microprocessor and its family support is chosen for the instrumentation proposed for the real time evaluation of railway track profiles. In this Chapter, the track profile measurement system's specifications, system architecture, and hardware details of the system which also include. system packaging have been described in detail.

4.2 SYSTEM SPECIFICATIONS

To make the system upward compatible and to have a proper system-to-user interface and also for having electrical compatability with the existing rail cars, the following specifications for the rail track monitoring system have been laid down.

4.2.1 Environmental Specifications

a. The system should be able to operate on + 24V DC battery system normally available on railway carriages.

b. It should be portable and should not require any air-conditioning.

c. It should be able to survive in the vibration environment normally obtainable on a coach floor.

4.2.2 Signal Processing Requirements and Real Time Nature of the Signal Processor

a) Track Parameters to be extracted: For every kilometer span of track length, absolute vertical profiles of left and right rails, absolute lateral profiles of left and right rails, dynamic cross level, centre line unevenness, dynamic gauge, centre line alignment and power spectral measurement of each of the track profile is to be evaluated by the system in real time (i.e. max. 30 seconds : the time taken for the rail car to traverse 1 Km at the envisaged maximum speed of 120 Km per hour).

b) Analog Signal Channels: The system is to process the outputs of four accelerometers mounted on the axle boxes for both horizontal and lateral profiles of the left and right rails. Hence the system should be able to sample, digitize and store them simultaneously from these four transducers for subsequent signal processing.

c) Conditioning of the Input Signals: The outputs of the accelerometer should be suitably conditioned to the compatible levels of the system input digitizer.

d) Number of Points Per Channel: The number of sample points for each channel is fixed to be 512 for every kilometre section of the track which corresponds to an approximate spatial sampling of 2 metres. These sampling instant pulses are generated by the wheel and its associated electronics.

e) Processing Time: Considering the maximum speed of rail car to be 120 Kilometres per hour, there is 30 seconds of time available for each Kilometre traverse. Hence for each Kilometre of track span, all the four channel signal processing and track categorization should be carried out within 30 seconds.

f) Machine to Operator Interface: The system should be capable of accepting the initial kilometre entry at the time of starting of the run and automatic updating of the kilometre should be done on command at the start of each kilometre of track. Automatic recording or printing of the results at the end of the run, as may be required, is also necessary.

g) Track to System Interface: Suitable accelerometers will be mounted on axle boxes to get the acceleration signals from the track and these signals will be properly

conditioned before feeding to the signal processor.

4.3 SYSTEM DESCRIPTION

The system architecture is shown in Fig. 4.1. It comprises of a general microcomputer system and a monitor on a bus and application dependent modules. It consists of the following modules:

a) 6800 processor based CPU module with monitor ROM, stack memory, serial I/O (ACIA), parallel I/O (PIA).

b) 16 K ROM module for program storage.

c) 16 K Dynamic RAM module.

d) Ram Refresher Module which generates all the timing pulses for Dynamic RAM module and address decoding circuitry.

e) 8K Static RAM module (optional)

f) Digital Hardware multiplier module.

g) Console Module which consists of a fifteen digit eight segment display and a Hex Keyboard.

h) Cassette and RS 232 C interface module.

i) 20mA Loop TTY interface and Baud rate clock module.

j) Floppy disc drive controller module (optional).

k) DMA controller module (optional).

l) Analog output device controller module (optional).

m) A/D converter module.

n) Analog multiplexer module.

In the above, the modules (a) to (k) are for configuring a general microcomputer system where as the modules (1) to (n) are for the envisaged specific application.

4.3.1 Microcomputer System Hardware

In this section, the modules which comprise the microcomputer system that will be used in the proposed system have been described. To make the system modular, double sided printed circuited boards of size $6\frac{1}{2}$" x $4\frac{1}{2}$" have been used for all the modules with a 44 pin edge connector contacts that go on to the system bus. The edge connector tongue designation with associated bus signals are given in Table 4.1 and the same will be followed to all the hardware modules except where mentioned.

4.3.1.1 CPU Module

The complete circuit diagram of the CPU module is shown in Fig. 4.3. This mainly consists of a CPU clock and decode circuitry, on board baud rate clock, processor, memory and I/O provisions.

a) CPU Clock and Decode Circuitry: Since the system uses M6800 microprocessor which uses a two phase clock (ϕ_1, ϕ_2), it is required to generate a two phase non overlapping clock that should run at V_{cc} level, according to the requirement of 6800 processor. The M6800 uses a 1MHz clock. To derive this a 4 MHz clock has

been used comprising of a 4 MHz crystal and a part of U1 (7400 IC), the output of which will be further divided by a factor of four using U2 (7473). The output of U2 will be a 1 MHz clock which will be buffered using U3 (7404), part of U1, U5 (7437) and two transistors Q_1, Q_2 (2N2222) to derive a nonoverlapping two phase clock. The decoding circuitry consists of U4 (7410) which generates the various timing signals, using Valid Memory Address (VMA) of the CPU, like VMA.$\emptyset_2$, $\overline{VNA.\emptyset_2}$, $\overline{RCS}$. The $\overline{RCS}$ is the chip select for the Read Only Memory (ROM) on CPU module where the 'Monitor' of the system actually resides. $\overline{RCS}$ is 'ANDED' configuration of $\emptyset_2$ clock, ROMEN1, ROMEN2 where ROMEN1 is ROM ENABLE1 and ROMEN2 is ROM ENABLE2. These signals are derived from Ram Refresher Module. ROMEN1 is a buffered version of R/$\overline{W}$ (Read/$\overline{write}$) and ROMEN2 will be in active state when the MSB address lines have got an Hex address of 'F'.

b) On Board Baud Rate Clock: This clock will be used in conjunction with Asynchronous Communication Interface Adapter (ACIA 6850) for serial I/O application. This is a simple astable multivibrator using U15 (555), a timer IC with a precise baud rate (300 baud) adjustment using a trimpot. If higher baud rates are required this on board baud rate clock should be disabled, and external clock should be connected to the system.

c) Processor, Memory and I/O: This section is the heart of the system consisting of 6800 microprocessor, 1 K R/W memory (a part will be used as stack memory), two Peripheral Interface Adapters (PIA 6820) for parallel I/o communication, an ACIA (6850) for serial I/o applications and a 1 K ROM space where the 'Monitor' ROM resides. The processor address, data and control lines are brought on to the edge connector contacts which go into the system bus. The PIA outputs are available on a 50 pin Eurocard type connector. The ACIA connections are brought on to a 14 pin DIP socket. For serial and Parallel I/O communications proper jumper assemblies should be used for the Eurocard type connector and DIP socket. In the system configuration that is used for this application, the DIP jumper assembly goes to the RS 232 and cassette interface module which in turn can be used for the Asynchronous communications like cassette, RS 232 C, Teletype etc. The memory mapping of the various I/O extensions and the memory on CPU module is shown in Table 4.2. The system reset is hard wired through a pushbutton switch to the reset pin (pin 40) of the processor (or tongue Z of the edge connector). The serial connector (14 pin DIP socket) signal summary is given at Table 4.3. After the power-on condition and the system reset, the monitor that resides in the ROM takes the control of the system. For its use, the monitor mainly uses the stack area on board.

4.3.1.2 16 K ROM Module

This module provides 16 K byte memory accessory for the control module. This module uses 2708 erasable/ programmable type ROM ICs. The ROM card may be configured into any quadrant of the processor's address space using jumper selection of the higher order bits. The ROM array is fully buffered from the microprocessor bus using low power buffer gates to assure minimum loading. The complete circuit diagram is given at Fig. 4.4. U1 and U2 are low power gates which isolate the capacitive loads of the ROM array from the system bus. A_0 to A_9 (the LSB address) are supplied directly to the 2708 address leads. A_{10}, A_{11}, A_{12} and A_{13} are buffered by U_3 and U_4 and used as inputs to a 74154 (one-of-16 decoder). This decoder generates the chip select signals to select one of the 2708's from the array. Address A_{14} and A_{15} are decoded to enable the data bus drivers U6 and U7. The state of A_{14} and A_{15} which enables the board is jumper selectable. The placement of these jumpers will determine where the ROM board appears in the processors memory map. Four jumpers are used to make the memory space selection. Only two of these jumpers are ever in place at any one time. Jumper insertion and the resulting memory map is shown below:

Memory locations occupied	Jumpers in place
$C000 - FFFF_{16}$	J1, J4
$8000 - BFFF_{16}$	J2, J3
$4000 - 7FFF_{16}$	J2, J4
$0000 - 3FFF_{16}$	J2, J3

4.3.1.3 16K Dynamic RAM Module

This RAM module is a high density memory accessary for the CPU module. The complete circuit diagram is shown in Fig. 4.5. This RAM module uses 4096 dynamic RAM IC. This module accepts the multiplexed addresses and timing signals from the RAM REFRESHER Module through the refresh bus and communicates data thru system bus. Each 4096 byte bank of RAM has an associated TTL gate to supply the ROW Address Strobe (RAS) and Chip Select (CS) signals to the eight IC's comprising that bank. The RAS decoding is used to minimise the quiescent power dissipation. This is achieved by giving the RAS signal only to the particular RAM bank selected. During the refresh cycle the REF signal from the refresher module forces RAS to be applied to all the memory ICs. During the refresh cycle, processor does not receive $\emptyset_2$, hence does not generate VMA.$\emptyset_2$. This signal is gated through CS to the RAMs so that all RAMs are deselected during a refresh cycle. When RAM refresher module is used, the onboard clock should be disabled by removing U2 on CPU module since the clock is

now supplied by refresher module. Half of U4 is used to detect a reference to the memory board. Pin 6 of this IC goes high when any of the four banks is to be read from or written into. The other half of U4 and part of U7 generate enable signals for a pair of Tri-state transceivers U5 and U6, which have high input impedance to minimize the loading on memories and processor data bus. Only one half of these transceivers is enabled at any one time. The half that is to be enabled is determined by the state of the R/W line from the refresher module. This R/W signal is also combined with the WRITE pulse from the refresher module to apply the write signal to the RAM array at the proper time in the memory timing cycle. Most of the U3 is used as a buffer to drive the capacitive load of the Column Address Strobe (CAS) pins on the RAM array where as U2 is used as buffer for the multiplexed addresses for the RAM array. Typical wave forms of the timing signals are shown in Fig. 4.6. These signals are buffered by TTL gates to minimize loading on the signals. The configuration of the RAM module in the processor's memory space is jumper selectable. Each 4096 bite bank may be configured independently at sixteen locations. The jumper termination with its associated address is given in Fig. 4.7. All the supply voltages are derived from the Ram refresher module.

4.3.1.4 Ram Refresher Module

This module is the link between the Dynamic RAM module and the CPU module. This provides the required timing and refreshing signals for dynamic RAM module and also buffers the address signals from CPU module. The complete circuit diagram is shown in Fig. 4.8.

IC1 comprises an 8 MHz crystal oscillator for the processor master clock which will be obtained by dividing the 8 MHz clock (pin 8 of IC1). When this module is to be used with CPU module the clock on CPU module should be disabled by removing the JK flip flop (IC2) on that module.

IC5 and IC6 (7495) form a divide-by-eight counter and memory clock generator. The signals shown in Fig. 4.9 with their waveforms are generated by this shift Register chain. The $\overline{\text{Row Address Strobe}}$ ($\overline{\text{RAS}}$), $\overline{\text{Coulmn Address Strobe}}$ ($\overline{\text{CAS}}$) and $\overline{\text{WRITE}}$ are the required timing signals for the dynamic RAM module (refer description of Dynamic RAM module); the processor clock $\overline{\emptyset}_2$ is supplied to the system bus through a Nand Gate (IC7) and is also used to operate the memory address multiplexer consisting of IC10 and IC13. These ICs are tri-state low power schottky devices used for minimizing the bus loading. IC2, IC3 and half of IC4 form a 32 μs timer which determines the refresh interval. When a refresh cycle is required, pin 12 of IC4 goes high; 125 ns into the next memory cycle pin 9 of IC4 goes high indicating that

the memory cycle is to be a refresh cycle. Refresh addresses from IC8, IC9 are supplied to the memory board through IC11 and IC12. The clock $\emptyset_2$ is suppressed during the refresh cycle. The REF signal is given to the memory card to signal it that a refresh cycle is in progress. The higher order bits of the address (A_{12}, A_{13}, A_{14}, A_{15}) are buffered by IC17 and are presented to a four-to-sixteen line decoder (IC14 74154). The output of this decoder is supplied to the RAM card where these will be used to select 4 K byte RAM banks. IC14 asserts $\overline{\text{RAMSEL}}$ when it decodes 'E' on the 4 highest address bits. The CPU module and interface modules use $\overline{\text{RAMSEL}}$ along with decoded address bits to generate chip select signals to activate the various PIAs, ACIAs etc. U15 asserts ROMEN2 when it decodes 'F' on the four highest address bits. ROMEN1 is the buffered version of R/W. Both ROMEN1 and ROMEN2 are used to enable the ROM on CPU module.

4.3.1.5 8 K Static Ram Module

Although packing density is low, the advantage with static RAMs is that refreshing is not necessary for these memories. A 8 K static RAM module using Intel 2114 (1 K x 4 bit) static RAM is also designed and the complete circuit diagram is shown in Fig. 4.10. In this all the address lines are buffered using low power schottky devices (to minimize bus loading) IC3, IC4, IC5 (74LS365). R/W, $\overline{\text{VMA}.\emptyset_2}$ are also buffered using these ICs. Data lines go to two

data bus transceivers (8833) that communicate with memory ICs. In this, there are two 4 K memory banks and the memory space allocation for these banks is jumper selectable. This is done using two 3-to-8 line decoders (74155) and two OR gates, and the outputs of these OR gates will form as chip select signal for a 1 K x 8 bit section in which two 2114s will be paralleled. The enabling of the data bus transceiver is done using the two bank select signals and R/W signal; R/W selects the mode of data bus transceiver i.e. transmit or receive. This module is also bus compatible following same tongue designation for the edge connector contacts.

4.3.1.6 Console Module

The console I/O module provides an economic means of directly inputting / outputting information between operator and system bus. The complete circuit diagram is given at Fig. 4.11. The module has got two main parts, display and keyboard. It has got a real time clock which interrupts the processor at regular intervals. The display consists of fifteen digits of eight segment LED readouts and the display codes are under full software control for maximum versatality. The refreshing feature of the display eliminates the need for latches and accomplishes this by turning on one digit at a time. The segments that are to be turned on will be determined by U1's "B side" bits. These are buffered by U5 and U6 which then drive the segments through current

limiting resisters R_1 through R_8. Each common cathode of the readouts is driven by U3 and U4 . one-of-16 coder driven by the display refresher counter U7 (7493). U7 is cleared by CA_2 of U1. No digit is on while U7 is clear. Then each successive digit is turned on, from left to right, by strobing U7's clock input by CB_2 of U1. Control register B of U1 is therefore configured so that when a new set of data is written into the PIA, CB_2 goes low in order to switch on the next column. The display can be turned off completely by control of CA_2 of U2. A 'zero' turns on the segment.

The PIA (U1) A-side with its internal pull up resistors is connected directly to the key switch. The common row terminals of each switch are connected to PA4-PA7 of U1. The common-column terminals are connected to PA0-PA3 of U1. If the A side is configured as an input, the key switch may be inputted by reading the eight bit A-side data. U8 (CD4060) is a real time clock (CMOS oscillator / ripple counter) which gives interrupts to the CPU at a rate of 75, 150, 300 or 1200 interrupts/second depending upon jumper J3 connection. Normally IRQ interrupts are generated by the console module. However a simple jumper modification allows NMI type interrupt. The display refresh routine will be as follows. The routine receives interrupts from the module configured for an interrupt interval consists of two sections: Initialization code performed once at reset time

and interrupt driven code which at interrupt time gets the display code for the next digit and outputs it to the display drivers. The memory space occupied by the module is selected via jumpers J1 and J2.

4.3.1.7 Cassette and RS 232 C Interface Module

This module provides the asynchronous communications to the CPU module through the use of a cassette deck or RS 232 console. The complete circuit diagram of this module is shown in Fig. 4.12. This module is 'not bus compatible' hence should be mounted separately, and uses the ACIA jumper assembly from CPU module.

WRITE Operation: During cassette write operations, the $\overline{\text{RTS}}$ lead from ACIA is made low. This makes the IC14 as a divide-by-four or a divide-by-eight counter. The divide constants depends upon the state of the TXD lead from ACIA. The 4800 Hz clock (300 baud clock) from cassette clock and TTY interface board is divided to 2400 Hz for a 'one' from the ACIA or 1200 Hz for a 'zero'. For high speed operation the 2400 Hz output of IC5 is supplied to the ACIA operating in a divided by 'one' mode. Low speed operation is achieved by selecting the 4800 Hz clock while operating the ACIA in the divided by sixteen mode. The timing chart for write operation is shown in Fig. 4.13.

As it can be seen from the timing chart for the high speed operation, 'ONE' is recorded as a single cycle

of 2400 Hz while 'zero' is recorded as one half cycle of 1200 Hz. In the low speed operation 'one' is recorded as eight cycles of 2400 Hz while zero is recorded as four cycles of 1200 Hz. Cassette write operations inhibit the output to the RXD lead by allowing IC5 to be clocked set thus forcing RXD to the ACIA to be held marking. Output to the RS232 console is also inhibited by IC8.

READ Operation: IC2 and most of IC1 and IC3 comprise a limiter/edge detector which squares the tape audio data and generates short positive pulses at the output of IC3 (pin 11) which is also the reset signal to the baud rate clock. Hence these pulses are used to synchronize baud rate clock to the incoming data. The clock signal (300 baud) from baud rate clock is supplied to a missing pulse detector comprised of IC4. Missing pulses indicating 'zero' data force the output of IC9 (pin 11) high which causes the RXD lead low for each zero bit detected. Read clock is recovered from the data by IC5. Phasing ambiguity is removed when reception of the first zero occurs. The missing clock pulse allows a low to be loaded into IC4 (pin 9). IC5 pin 5 is constrained to become high in the next transition of the recovered clock so that the succeeding bits will be centre sampled by ACIA. Input from the RS232 console is locked out by the data multiplexer comprised of IC6 and IC9. Output to the console is inhibited by IC8. Read timing signals are shown in Fig. 4.14.

This cassette and RS232 interface module is also provided with a logic to changeover the connections between RS232 and TTY. Since TTY uses the same ACIA jumper assembly this logic is necessary which comprises of IC9, IC10, IC11, IC12, IC13 as shown in the diagram. The RS232 interface comprised of IC8, IC15 (MC1488 line driver) and a transistor Q_1 (2N 2222) and D_3.

4.3.1.8 Baud Rate Clock and TTY Interface (20mA Loop)

This module provides the necessary baud rate clock for asynchronous communications and also has got a TTY interface (20mA loop) circuitry. The circuit diagram is shown in Fig. 4.15 and this module also not bus compatible. Hence should be mounted separately.

Baud Rate Clock: This comprises of a crystal controlled 307.2 KHz clock derived by two inverters (IC5) and three delay type flip flops (IC1, IC2) which will be used by cassette interface module. This master clock is further divided by IC3, IC4 which form a counter and give standard terminal baud rates viz. 150, 300, 600, 1200, 2400, 4800 and 9600. These baud rates from 150 to 9600 baud are switch selectable using a DIP type switch. These switch selected rates are supplied to the ACIA via a clock multiplexer circuit comprised of IC6 and IC9 on cassette interface module.

TTY Interface (20 mA loop) : The necessary TXD, RXD signals for serial transmission of this TTY interface are derived from a simple switch over logic between RS232-TTY on cassette interface module. The baud rate will be fixed at 300 baud by connecting the on board baud rate clock on CPU module to the TXC, RXC clocks. This is also done by the switch over logic. The TXD signal is buffered by a NAND gate (IC6) and is coupled to a opto isolator (IC7) which in turn supplies the necessary current for the TTY loop. Similarly TTY output signals are coupled through a opto isolator (IC8) and a NAND gate (IC6) to the RXD terminal of ACIA. The TTL logic levels are converted to a current loop by the opto isolators.

4.3.2 Microcomputer System Configuration and Software

The above described modules configure a complete microcomputer system with the input/output provisions for a cassette deck, RS 232 console or a Teletype and a console module with a Hex key board and fifteen eight segment LED display. The basic system memory map will be as follows:

a) CPU Module

Hex address	Component
0000 - 03FF	RAM
ED80 - EDFF	RAM (stack memory)
EE08	U9 ACIA control/status register
EE09	U9 Data Register

EE10	U8 (PIA-1) output register A
EE11	U8 output register B
EE12	U8 control register A
EE13	U8 control register B
EE20	U7 (PIA-2) output register A
EE21	U7 output register B
EE22	U7 control register A
EE23	U7 control register B
FC00 - FFFF	Read only memory

b) Console module base address at $EE80_{16}$

c) 16K Dynamic Ram memory allocation from $5000 - 9FFF_{16}$

d) 8K Static Ram memory allocation from $2000 - 3FFF_{16}$

e) 16K ROM module memory allocation from –

f) A/D converter PIA address $EE50_{16}$

Software

The system software supports a ROM based 1K-byte loader/monitor/debugger programme for 6800. The monitor resides in 2708 ROM and is on CPU module. The monitor supports the following commands when entered from a teletype or video terminal.

Features of the monitor: The following are the monitor commands.

M - Display/change memory

I - Display/change instruction

R - Display processor registers

CNTL-A - Display/change accumulator A

CNTL-B - Display/change accumulator B

CNTL-C - Display/change condition code register

CNTL-P - Display/change program counter

CNTL-X - Display/change index register

G - Go to user program

S - Single step user program

D - Dump memory

P - Punch loader compatible tape

L - Load memory

The monitor also supports

- free-format hexadecimal input
- insertion and deletion of break-points
- interrupt vectors

Free-format means the leading zeros can be omitted when entering hexadecimal address, data, OP codes, or the first or second byte of an operand. After the power-on condition and pushing the reset switch reverts system control to monitor. Then the monitor responds with a carriage return (CR), line feed (LF) and an asterisk (*). The '*' indicates that monitor is in the command mode and ready to accept any of the monitor commands. RESET should be used whenever power is first applied or system control to be returned to monitor.

Interupt Vectors

a) Software Interupt: If the processor encounters an SWI instruction, monitor responds by pushing the MPU registers onto the users stack and then vectors to memory address $EDFA_{16}$. Memory addresses EDFA, EDFB and EDFC are reserved for the users SWI handler or a jump instruction to the SWI handler at another memory location.

b) Interrupt Request: If the IRQ line is pulled low and the interrupt mask bit is not set, monitor responds by pushing the MPU registers onto the user stack and then vectors to memory address $EDF7_{16}$. Memory locations EDF7, EDF8, and EDF9 are reserved for the user IRQ handler, or a jump instruction to the IRQ handler at another memory location.

4.3.3 The Specific Application Dependent Hardware

For the track profile measurement, since the signals are in analog form A/D converter is necessary for digiting the input signals. Hence the following modules are used:

a) A/D convertor module
b) Analog multiplexer module

4.3.3.1 A/D Converter Module

The complete circuit diagram of this module is shown in Fig. 4.16 which is configured using 8700 series A/D converter. In this, the eight bits of converted analog

data are presented to PIA peripheral bits PA7 to PA0. The most significant bit is PA7 and least significant bit is PA0. The A-side control lines give some freedom of control over the ADC. The ADC may 'free run' or convert only when required under program control. An interrupt can be specified at the end of the conversion cycle or programmed I/O techniques can be used. To sample a specific channel the following routine is adopted. The routine is in two parts: initialization routine called only once at the start of the program and a sampling routine called when a sample is to be taken.

The memory space occupied by the module is selected via jumpers J1 and J2.System gain may be adjusted using Pot P11 on A/D converter module. The adjustment of this Pot allows the ADC to accept and convert voltages from a 1 volt to 20 volt interval. Offset Pot P10 allows the ADC to accept either unipolar or bipolar signals. If bipolar signals are chosen the inherent coding is offset binary. If two's complement coding is needed, the conversion is done in the software.

4.4 SYSTEM OPERATION

The signal processor for its operation, mainly uses the feature of interrupt vectors of the monitor. The console module works as a control panel with some keys marked as "RAT (Rail Track) monitor", "KM Scan", and "STOP". After the power on condition and system reset, the monitor takes over the control and waits for an input action. The console module, as described in the description of system hardware, gives regular interrupts. After the first interrupt is encountered, the monitor vectors to the one of interrupt vector address location. Since the console module is configured to NMI type interrupt the monitor vectors to the address location EDFD where EDFD, EDFE, EDFF are reserved for interrupt routine handler or a jump instruction to the IRQ handler. This routine tests the key pressed and if it is RAT monitor key, then the routine branches to the track monitoring routine, otherwise it waits until the key is depressed as mentioned in the operator-system operation interface flow diagram (Fig. 4.2). After the RAT monitor key is depressed, the track monitoring routine displays the word "FEED" on the console module LED display. Then the initial kilometer is entered and the routine performs the initialization of multiplexer and A/D modules. Data memory is divided into two buffer areas. When the data is collected, sampled and being stored at one buffer, the data processing on the other buffer is simultaneously performed. After 1 Km

track span data is evaluated, the results are buffered into the output data memory and cleared. Then next Km data is stored in this data buffer, processing of the data already stored in the other buffer is performed. Thus, the data buffers are alternatively used for data collection and processing.

The complete system operation is as follows. After the initialization of the multiplexer and A/D modules, the routine waits for the keyboard action, and checks whether Km Scan Key is depressed or not. This key is depressed at the start of the run, where the data is to be collected for the processing. If the Km Scan key is pressed, the routine increments the Km (present Km) and enables the data sample pulse generated by the rail wheel and its associated electronics. During the first Km of run, only data will be collected and stored in buffer 1 since no processing of the data is involved. In the next Km, data is stored in buffer 2 while the data in buffer 1 is processed for track evaluation purpose. The data processing scheme involves in finding the track profile of both left and right rails by filtering and double integrating the conditioned output of accelerometer. Then power spectral measurement of the track profile is done and the results are stored in the output data buffers, and after this the buffers will be cleared. In the third Km run, the data is stored in buffer 1 while the data in

buffer 2 is being processed. This alternate cycling between the data buffers is continued until the "Stop" key is depressed. After the depression of stop key, the routine disables the sample data pulse from the rail wheel, but continues processing of the data for the last Km. Then depending upon the status of the print key i.e. if Print Key is depressed or not the results will be printed on a printer and the control is taken over by the track monitoring routine for the next run. If the suspension of the activity of the processor is required at any moment, the system reset should be used for this purpose.

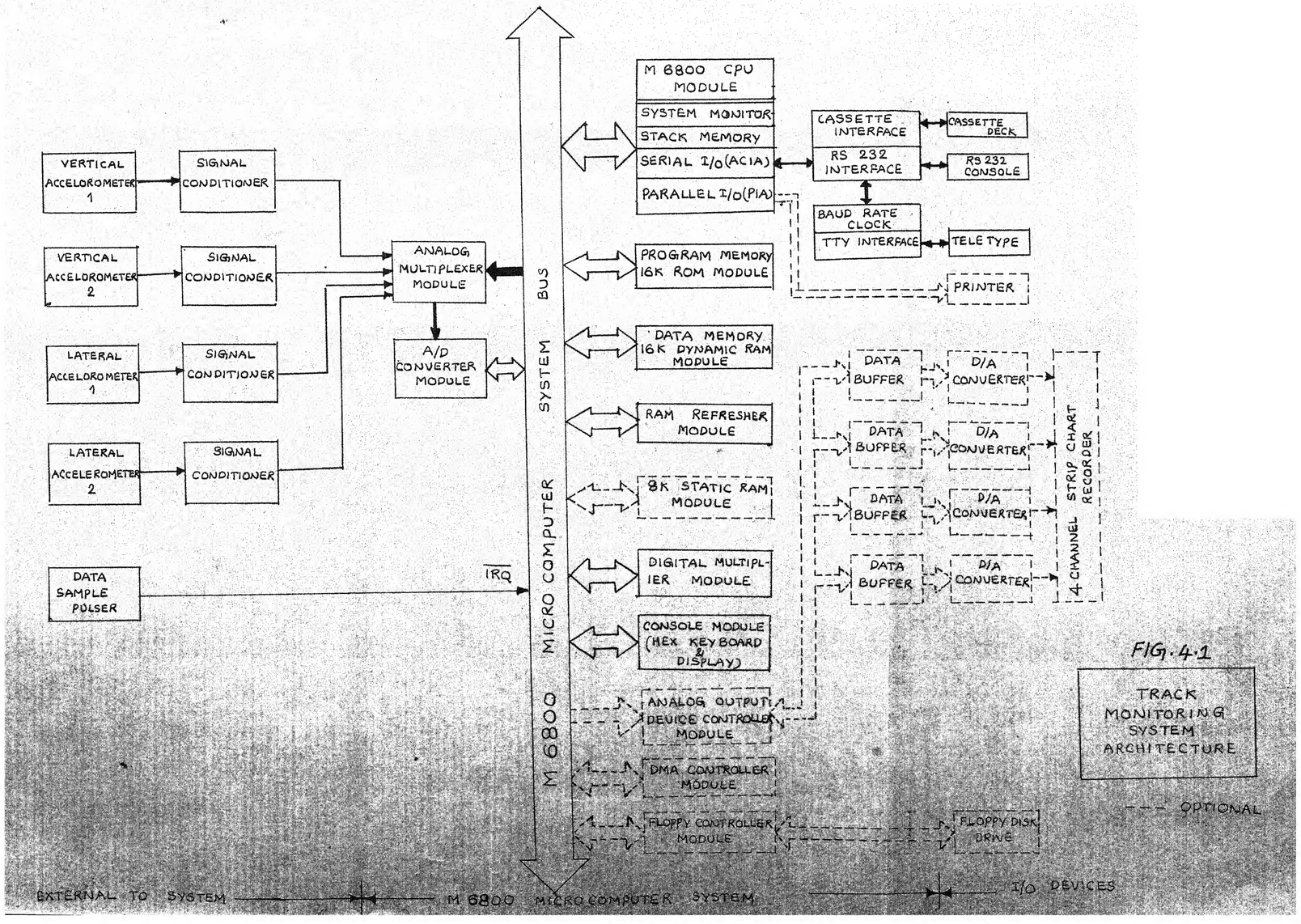


FIG. 4.1

TRACK MONITORING SYSTEM ARCHITECTURE

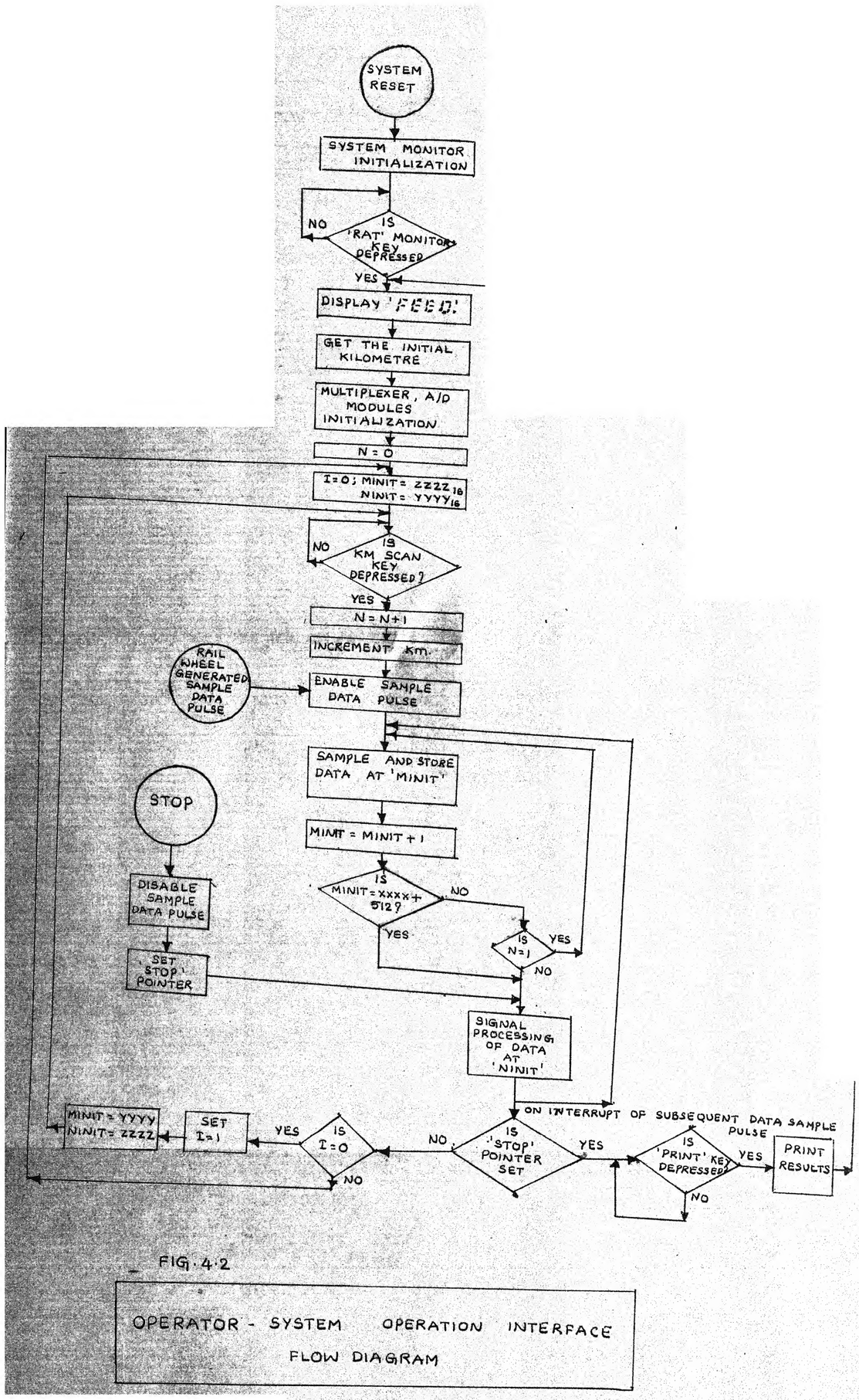
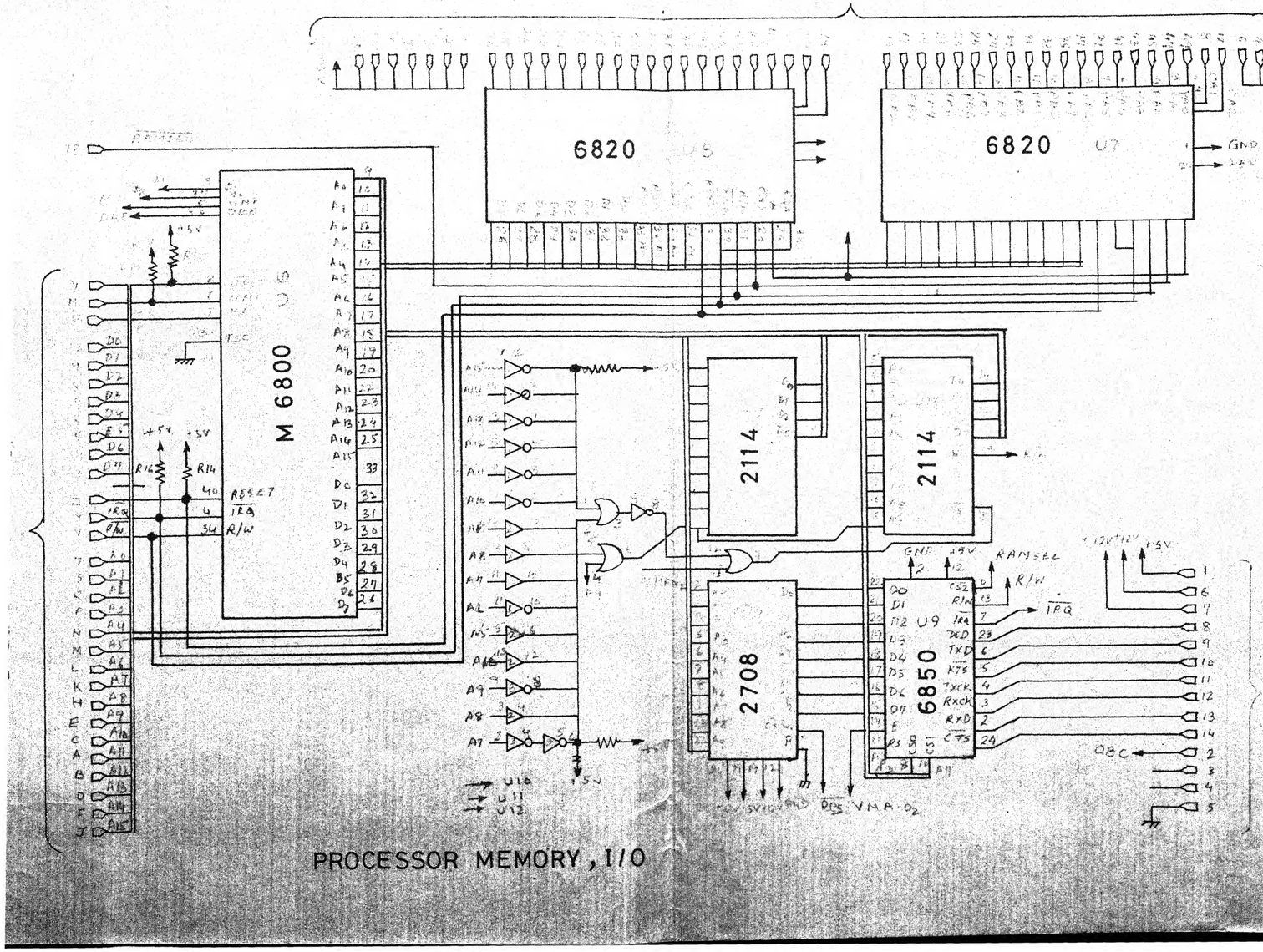
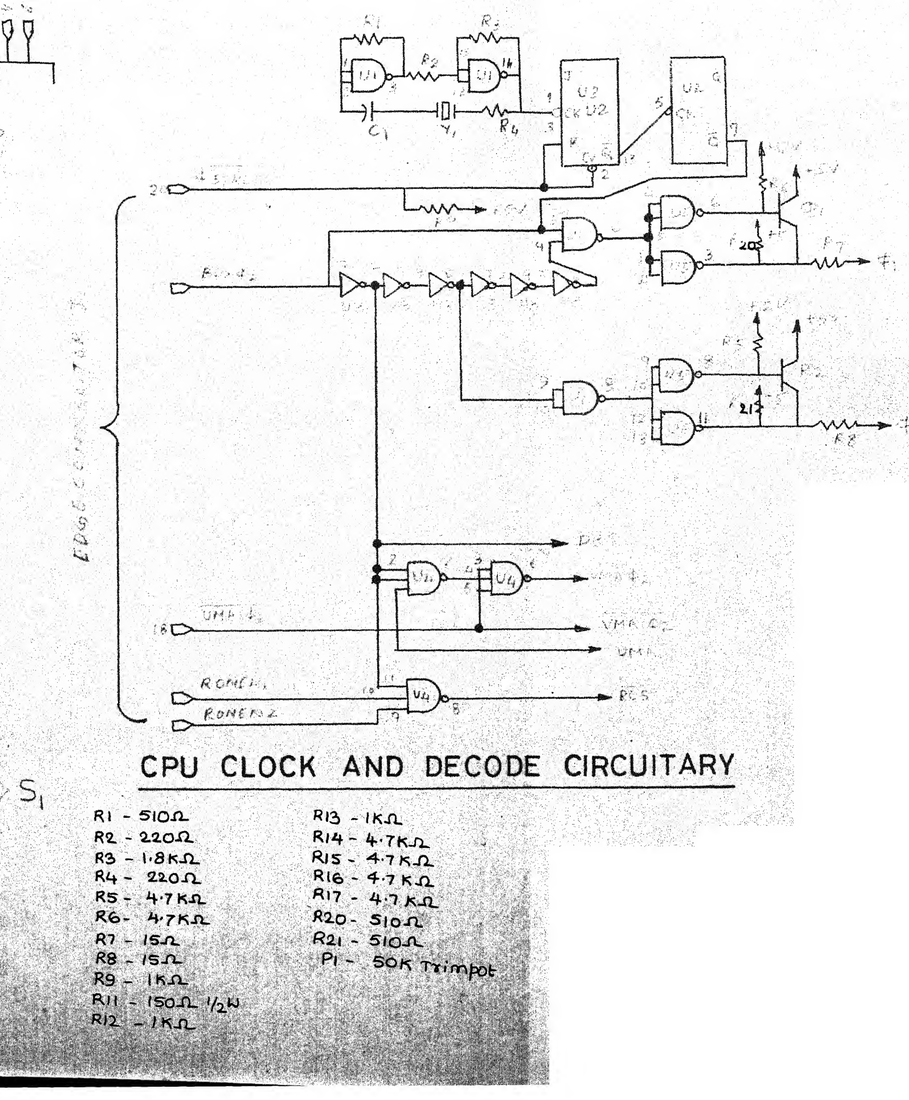


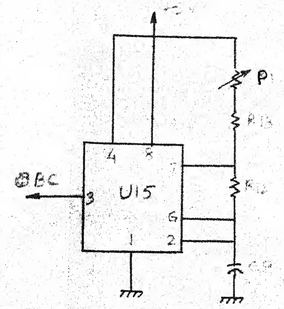
FIG. 4.2

OPERATOR - SYSTEM OPERATION INTERFACE FLOW DIAGRAM

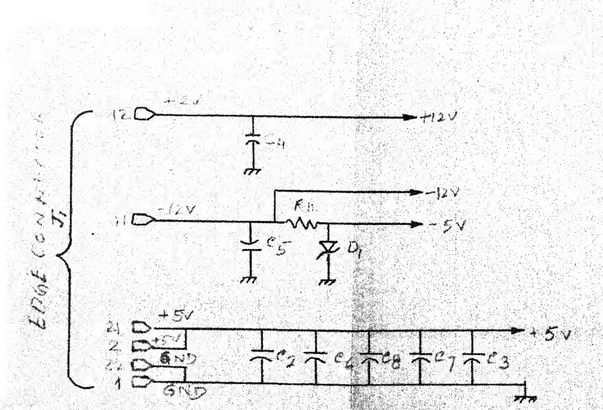




R1 - 510Ω
R2 - 220Ω
R3 - 1.8KΩ
R4 - 220Ω
R5 - 4.7KΩ
R6 - 4.7KΩ
R7 - 15Ω
R8 - 15Ω
R9 - 1KΩ
R11 - 150Ω 1/2W
R12 - 1KΩ
R13 - 1KΩ
R14 - 4.7KΩ
R15 - 4.7KΩ
R16 - 4.7KΩ
R17 - 4.7KΩ
R20 - 510Ω
R21 - 510Ω
P1 - 50K Trimpot



ON BOARD BAUD RATE CLOCK

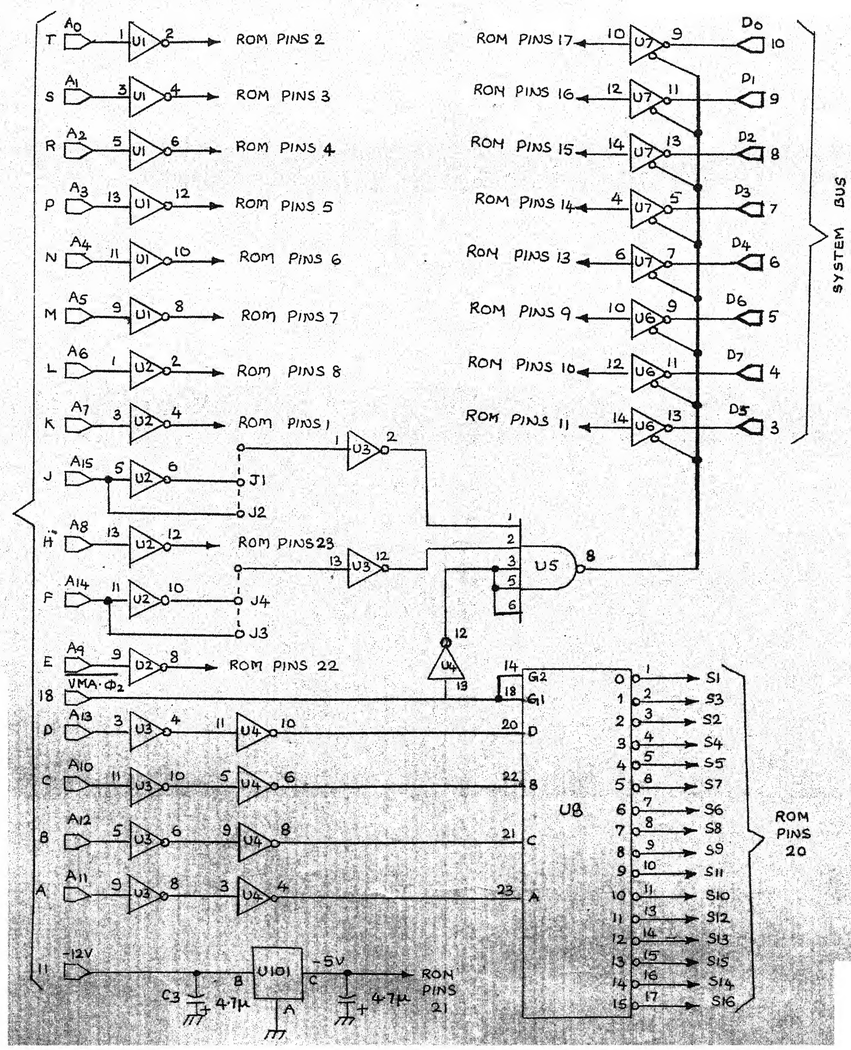


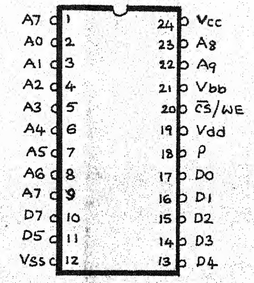
ON BOARD POWER DISTRIBUTION

U1 - 7400
U2 - 7473
U3 - 7404
U4 - 7410
U5 - 7437
U6 - 6800
U7 - 6820
U8 - 6820
U9 - 6850
U10 - 74LS05
U11 - 74LS07
U12 - 74LS05
U13 - 2114
U14 - 2708
U15 - 555
Q1, Q2 - 2N2222
Y1 - 4MHz crystal

FIG-4.3

M 6800 BASED SINGLE BOARD COMPUTER
6800 SYSTEM / CPU





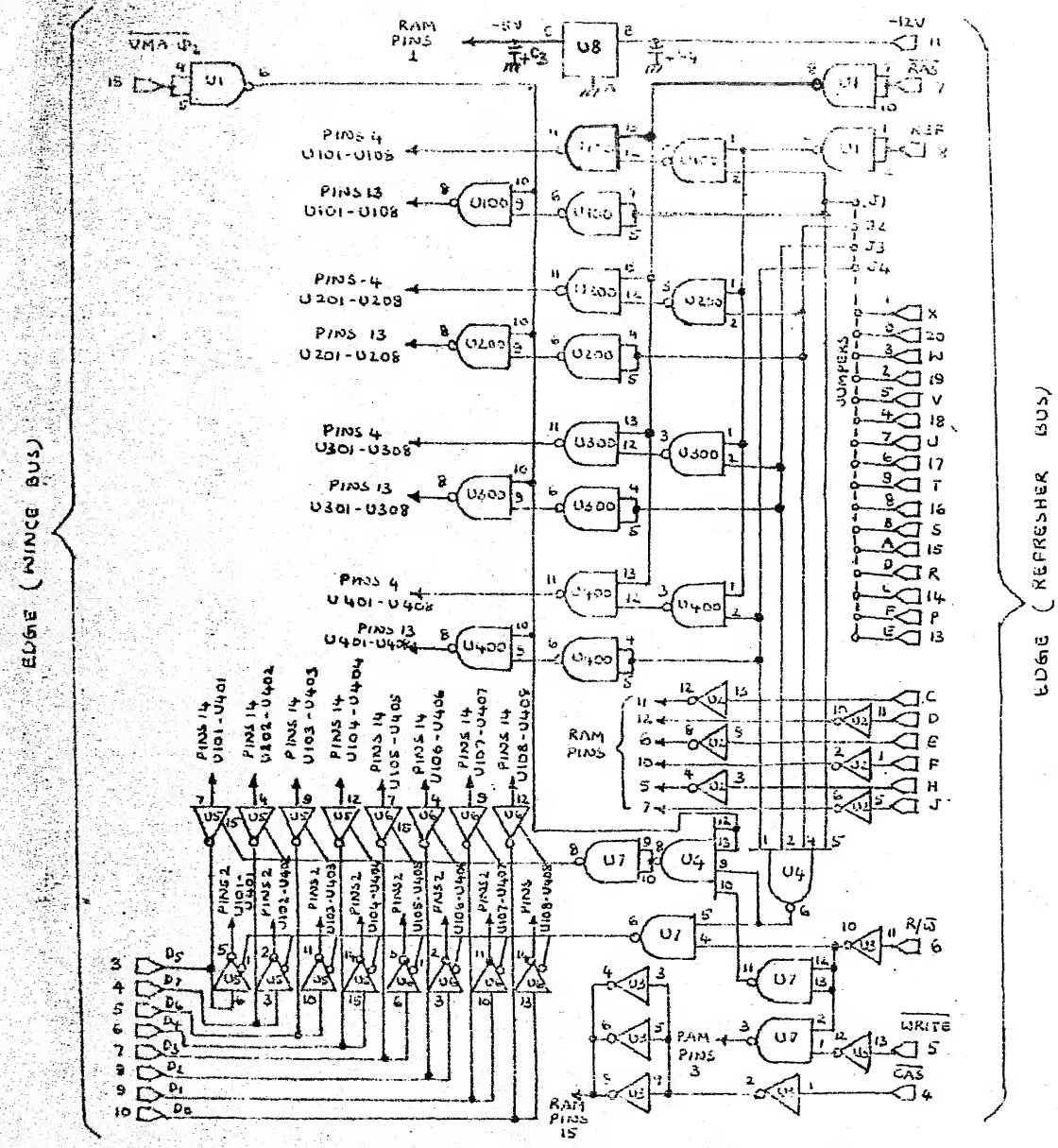
2708 PIN ASSIGNMENT

Vss = 0V.
Vcc = +5V
Vbb = -5V
Vdd = +12V
P = 0V.
CS/WE = 0/+5V } READ ONLY MODE

U1 74 L04
U2 74 L04
U3 74 C04
U4 74 L04
U5 74 L30
U6 74 LS 368
U7 74 LS 368
U8 74154
U101 7905

FIG·4·4

16K ROM MODULE



U1	7400
U2	7404
U3	74 HC+
U4	7420
U5	8T26
U6	8T26
U7	7400
U8	79 L05
U100	7400
U101 - U108	4096
U200	7400
U201 - U208	4096
U300	7400
U301 - U308	4096
U400	7400
U401 - U408	4096

FIG. 4.5

DYNAMIC RAM MODULE

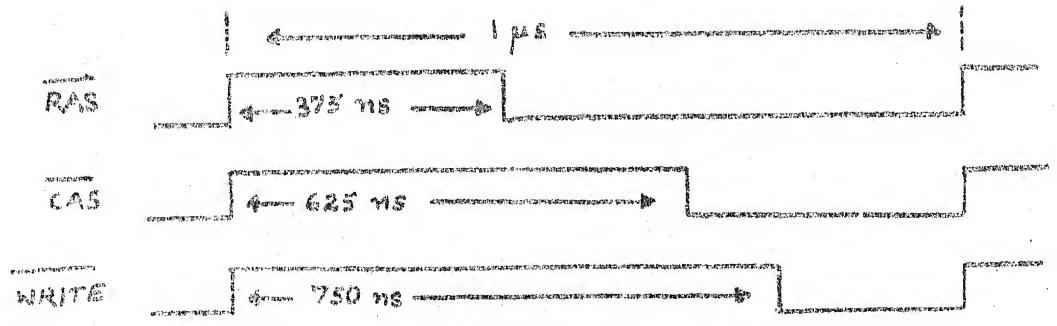


FIG: 4.6. TIMING SIGNALS FOR RAM MODULE

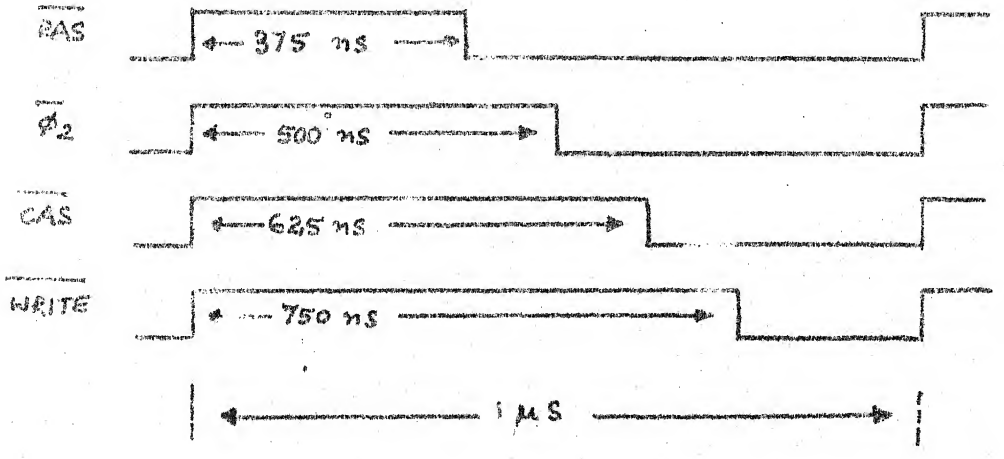
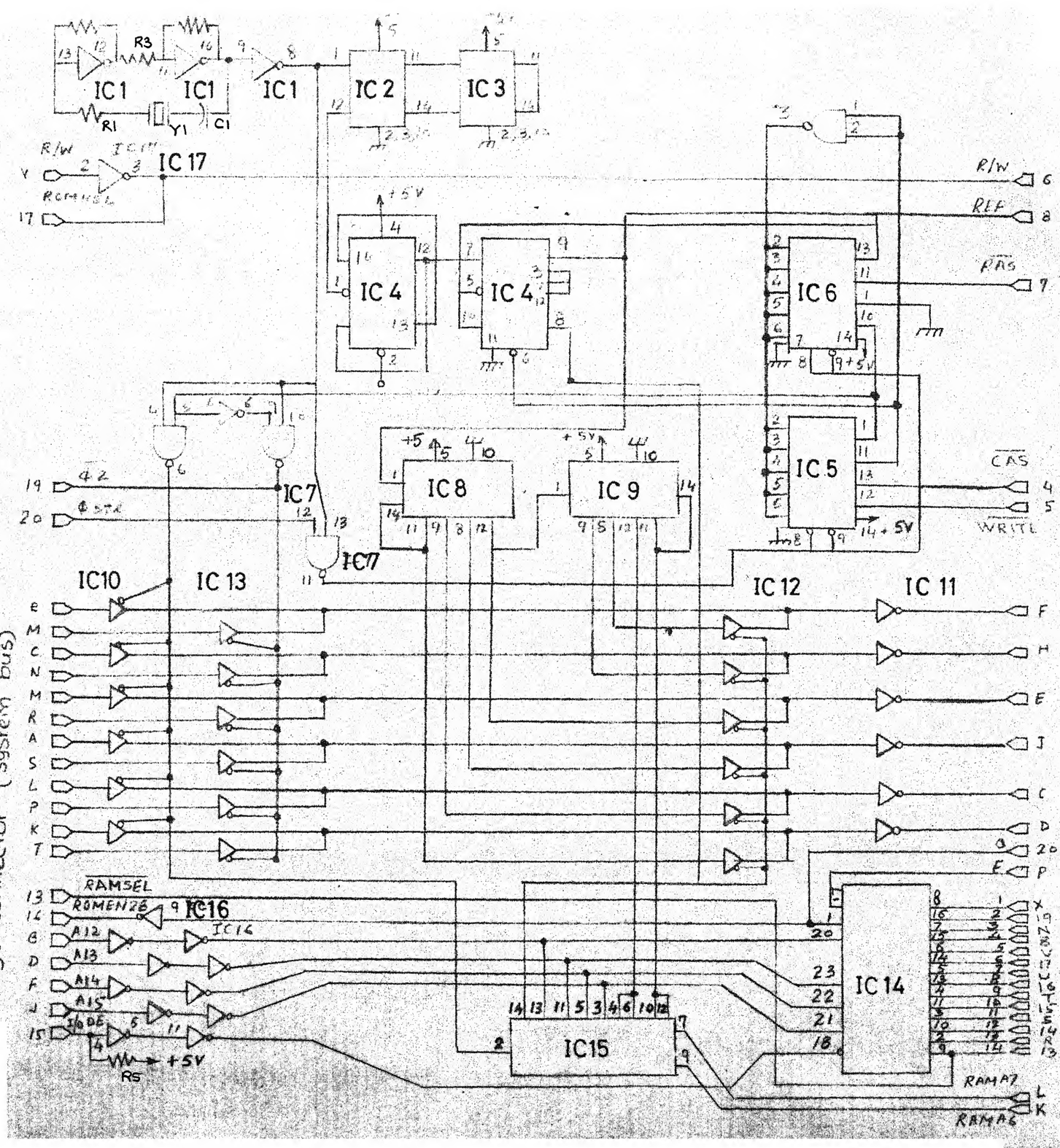


FIG. 4.7. TIMING SIGNALS OF RAM REFRESHER MODULE

Fig. 4.7 Jumper Termination and Associated Address for RAM Module

Jumper Termination	Memory Space
0	0000 - 0FFF
1	1000 - 1FFF
2	2000 - 3FFF
3	3000 - 3FFF
4	4000 - 4FFF
5	5000 - 5FFF
6	6000 - 6FFF
7	7000 - 7FFF
8	8000 - 8FFF
9	9000 - 9FFF
A	A000 - AFFF
B	B000 - BFFF
C	C000 - CFFF
D	D000 - DFFF
E	E000 - EFFF
F	F000 - FFFF



IC1 - 7404
IC2 - 7493
IC3 - 7493
IC4 - 7473
IC5 - 7495
IC6 - 7495
IC7 - 7400
IC8 - 7493
IC9 - 7493
IC10 - 74 LS 365
IC11 - 7404
IC12 - 74 LS 365
IC13 - 74 LS 365
IC14 - 74154
IC15 - 74153
IC16 - 7404
IC17 - 74 LS 365

R1 - 470Ω
R2 - 1KΩ
R3 - 470Ω
R4 - 4·7KΩ
R5 - 1KΩ
C1 - 35 pf
Y1 - 8 MHZ Crystal

FIG. 4.8

RAM REFRESHER MODULE

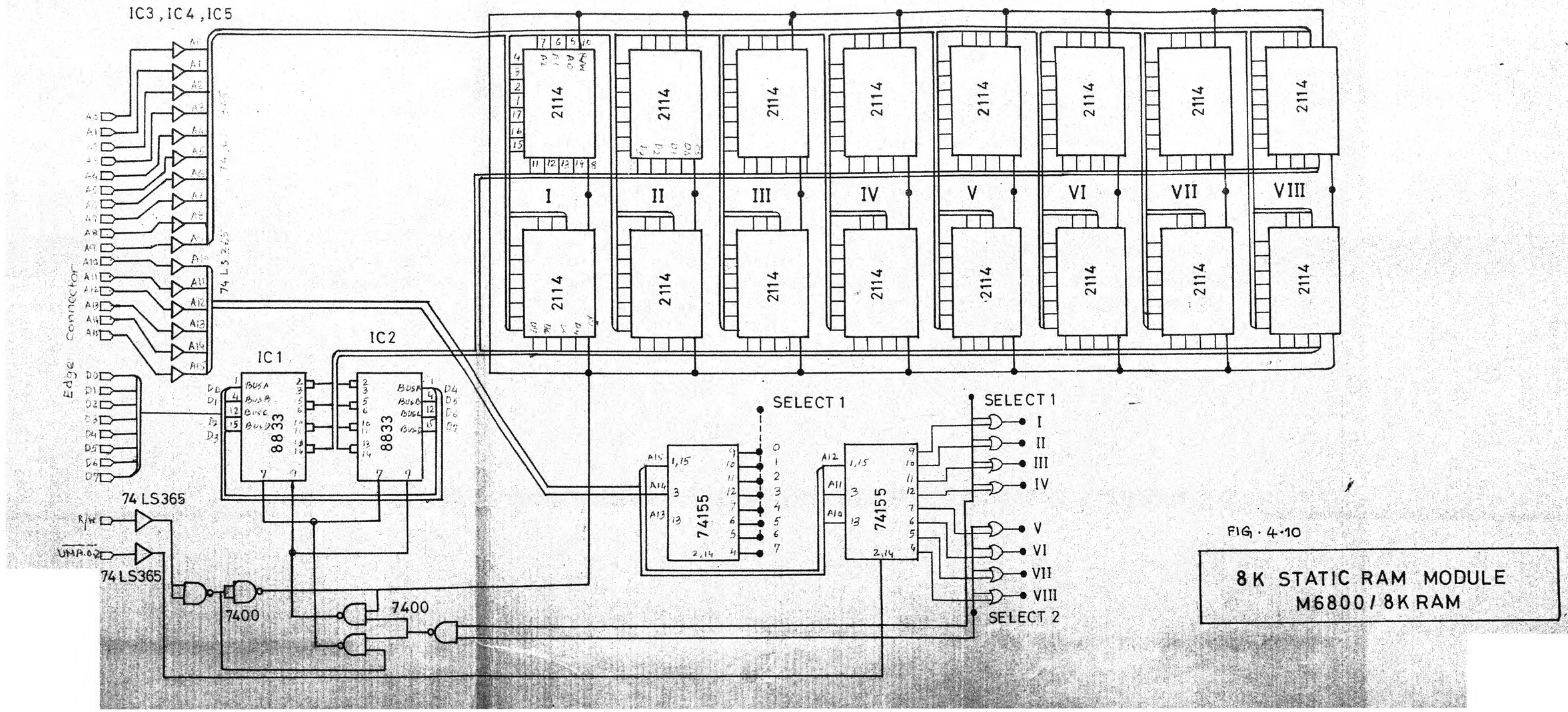
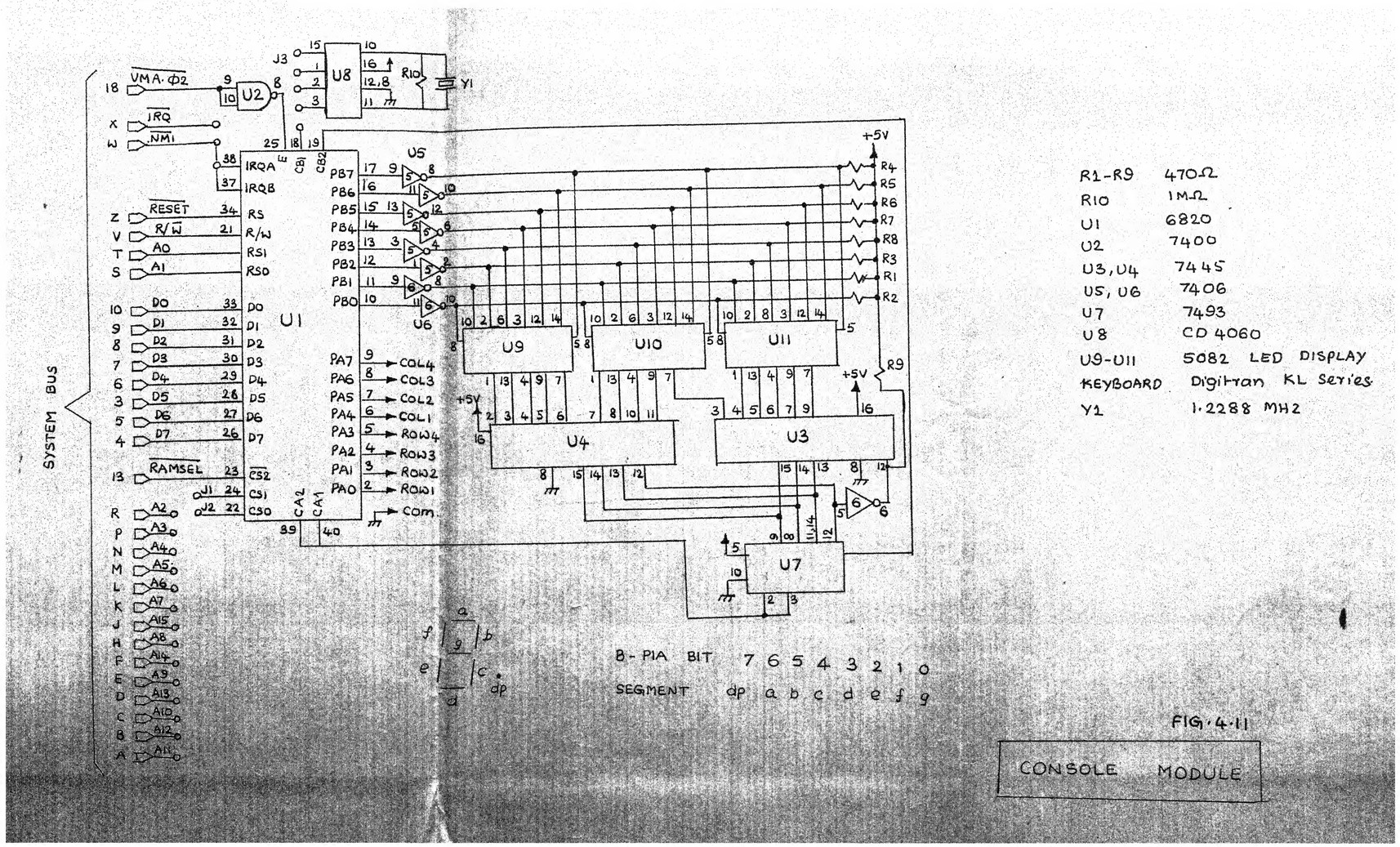


FIG · 4·10

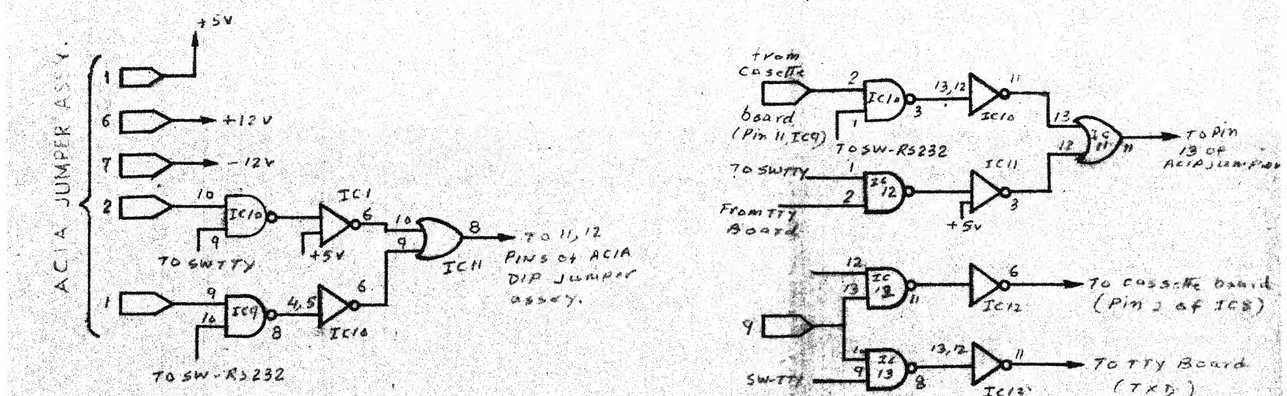
8K STATIC RAM MODULE
M6800 / 8K RAM



R1-R9	470Ω
R10	1MΩ
U1	6820
U2	7400
U3, U4	7445
U5, U6	7406
U7	7493
U8	CD 4060
U9-U11	5082 LED DISPLAY
KEYBOARD	Digitran KL series
Y1	1.2288 MHz

FIG. 4.11

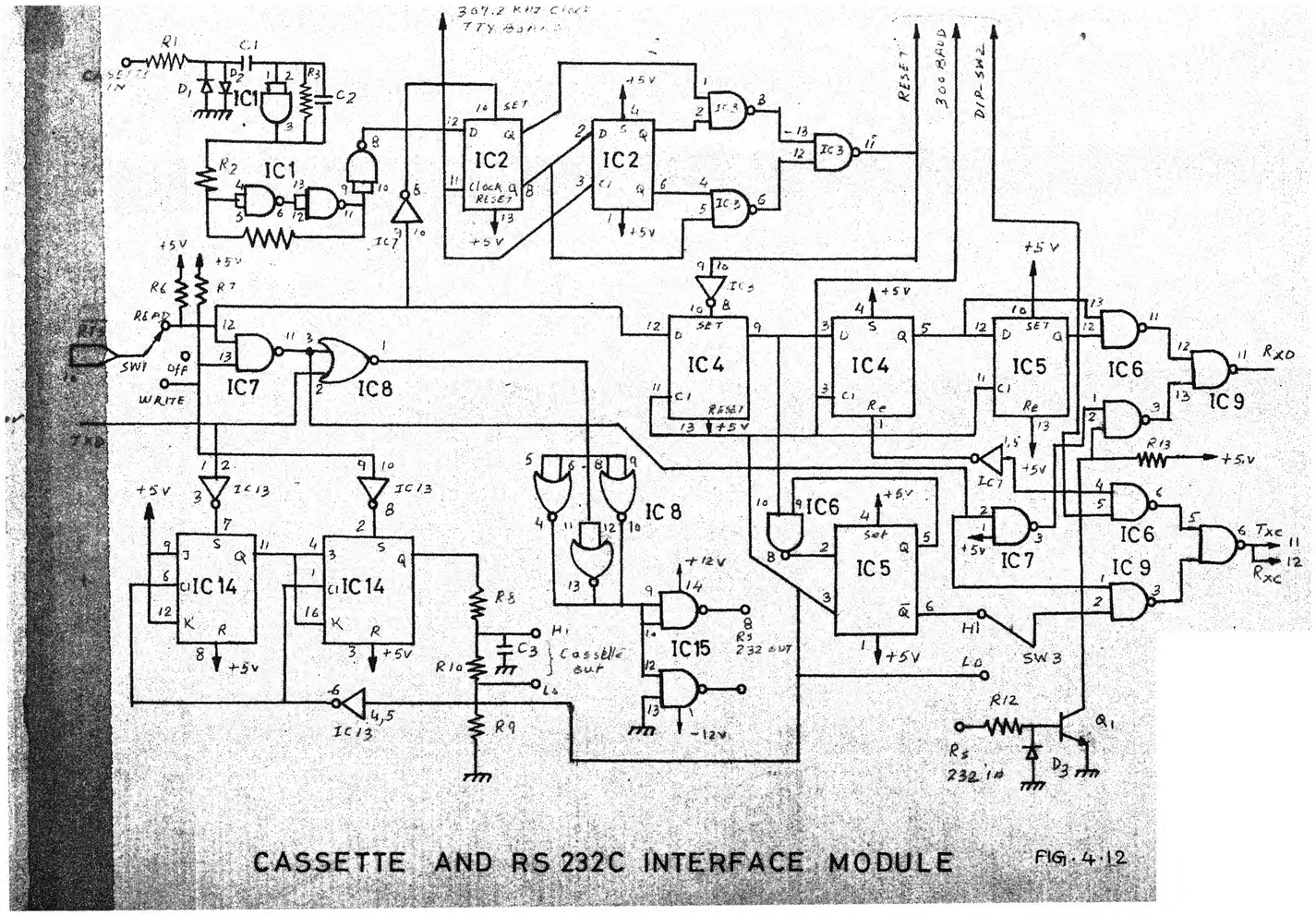
CONSOLE MODULE



RS 232 TTY CHANGE OVER LOGIC

IC1	74C00	IC9	74C00
IC2	74C74	IC10	74C00
IC3	74C00	IC11	74C86
IC4	74C74	IC12	74C00
IC5	74C74	IC13	74C00
IC6	74C00	IC14	74C76
IC7	74C00	IC15	74150 or MC1488
IC8	74C02		

R1 - 100KΩ		R11 - 1MΩ	
R2 - 100KΩ		R12 - 10KΩ	
R3 - 1MΩ		R13 - 10KΩ	
R5 - 1MΩ		C1 - 0.1 μf	
R6 - 10KΩ		C5 - 0.01 μf	
R7 - 10KΩ		C6 - 4.7 μf	
R8 - 100KΩ		SW1 - SPDT Centre Off	
R9 - 1KΩ		SW3 - SPDT SWITCH	
R10 - 10KΩ			



CASSETTE AND RS 232C INTERFACE MODULE FIG. 4.12

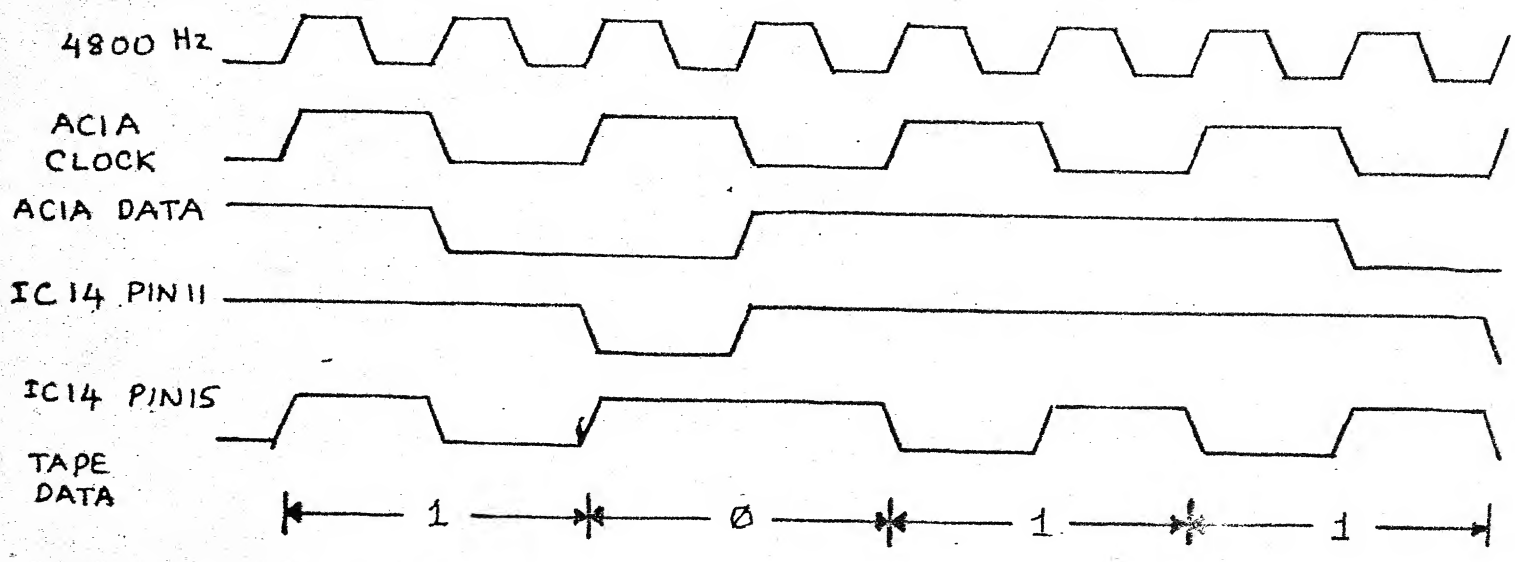


FIG. 4.13. TIMING CHART FOR WRITE OPERATION

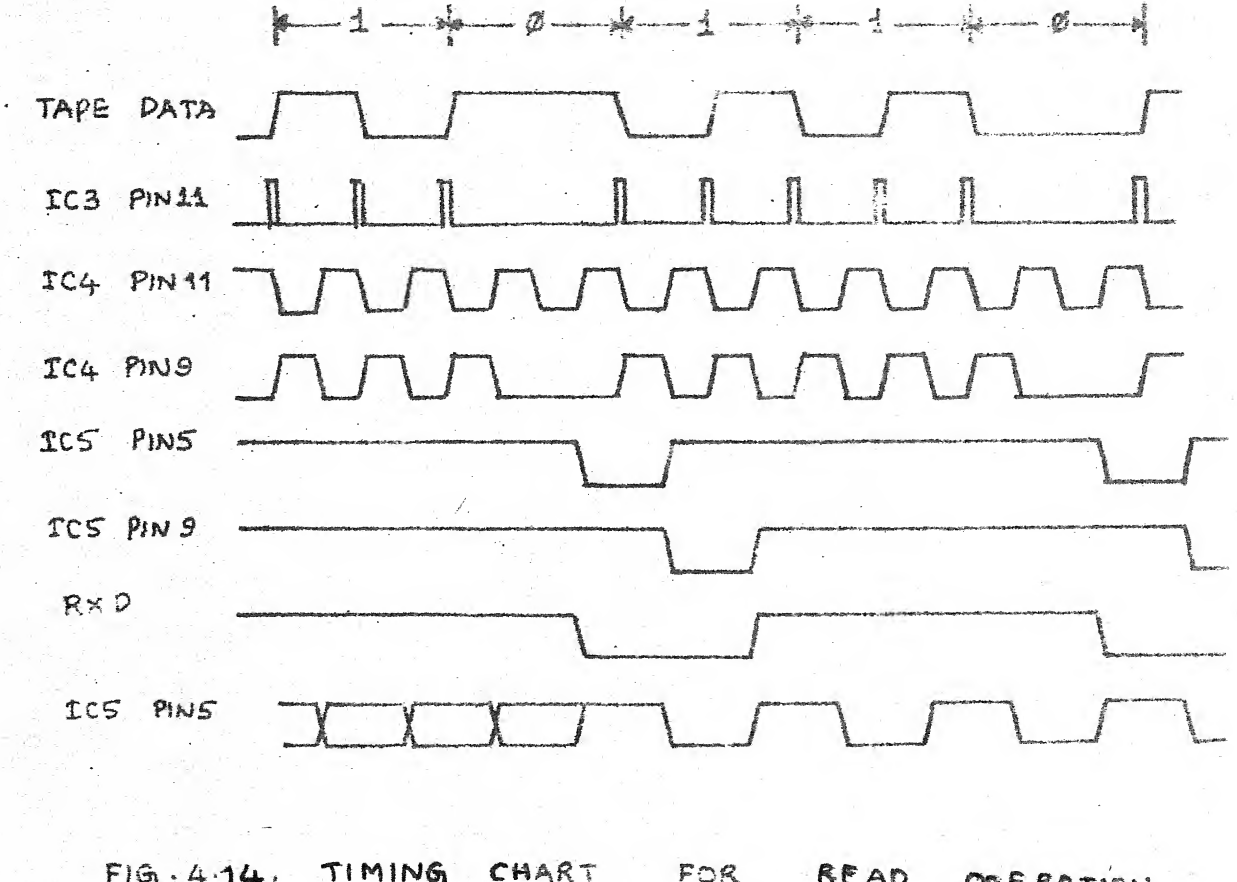
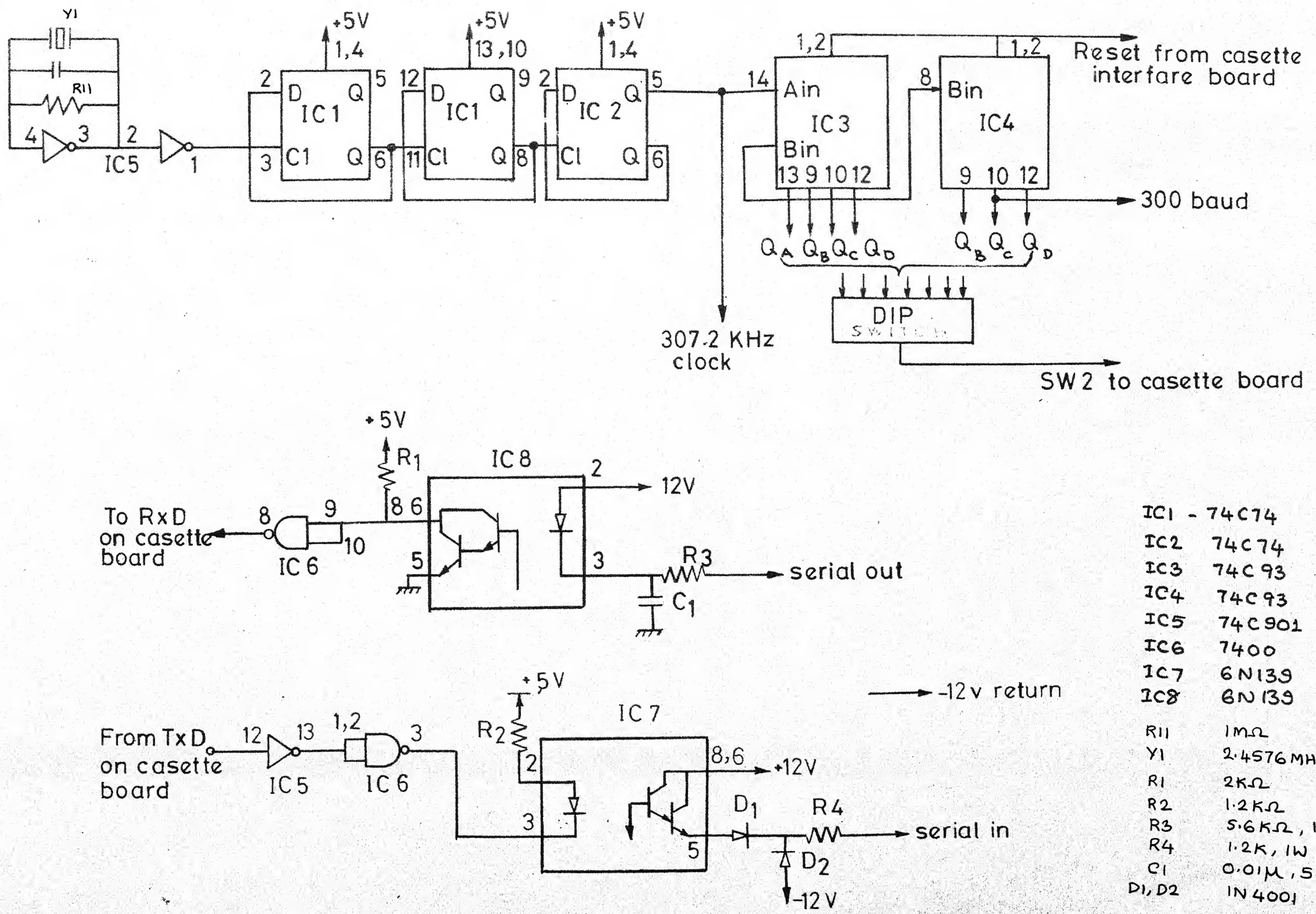
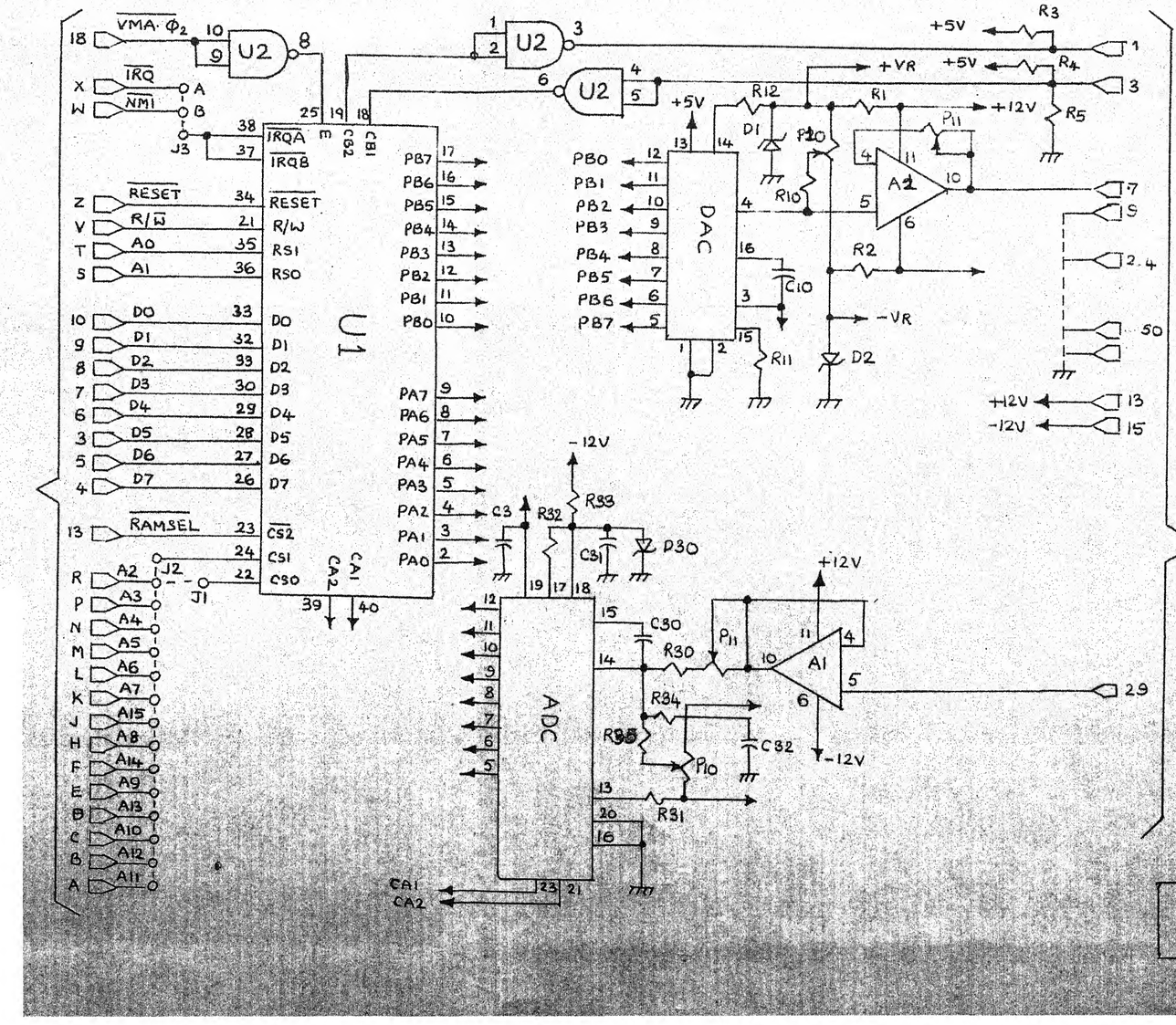


FIG. 4.14. TIMING CHART FOR READ OPERATION



IC1	74C74
IC2	74C74
IC3	74C93
IC4	74C93
IC5	74C901
IC6	7400
IC7	6N139
IC8	6N139
R11	1MΩ
Y1	2·4576 MHz
R1	2KΩ
R2	1·2KΩ
R3	5·6KΩ, 1W
R4	1·2K, 1W
C1	0·01μ, 50V
D1, D2	IN4001

FIG. 4·15 CASETTE CLOCK AND TTY INTERFACE



C1, C2	4.7μf 10V
C3	0.1μ 50V
U1	6820 PIA
U2	7400
D1, D2	IN751
R1, R2	150Ω 1/2W
R3	4.7K
R4	220Ω
R5	330Ω
FOR DAC	
R10	4.75K 1%
R11	4.7K
R12	4.75K 1%
P10, P11	20K 10T cermet
C10	35 pf
DAC	MC 1408L8
A1	741
FOR ADC	
R10	499K 1%
R30	100K 1%
R31	249K 1%
R32	100K
R33	150Ω, 1/2W
R34	100Ω
P10	20K 10T cermet
P11	2M 10T cermet
C30	68 pf
C31	0.1μ, 50V
C32	270 pf
A2	741
D30	IN751
ADC	8, 10 or 12 bit ADC

FIG. 4.16

A/D CONVERTER MODULE

Table 4.1 Edge Connector Signal Summary

Component Side		Wiring Side	
Tongue Designation	Signal	Tongue Designation	Signal
1	Gnd	A	A11
2	+5V	B	A12
3	D5	C	A10
4	D7	D	A13
5	D6	E	A9
6	D4	F	A14
7	D3	H	A8
8	D2	J	A15
9	D1	K	A7
10	D0	L	A6
11	-12V	M	A5
12	+12V	N	A4
13	$\overline{\text{RAM SELECT}}$	P	A3
14	-	R	A2
15	-	S	A1
16	ROM ENABLE 2	T	A0
17	ROM ENABLE 1	U	B.A
18	$\overline{\text{VMA} \cdot \emptyset_2}$	V	$R/\overline{W}$
19	$\overline{\text{BUS } \emptyset_2}$	W	$\overline{\text{NMI}}$
20	$\overline{\emptyset \text{ STRETCH}}$	X	$\overline{\text{IRQ}}$
21	+5V	Y	$\overline{\text{HALT}}$
22	GND	Z	$\overline{\text{RESET}}$

TABLE 4.2 CPU Module Memory Mapping

Hex Address	Component
0000 - 03FF	RAM
ED80 - EDFF	RAM (Stack Memory)
EE08	U9 ACIA Control/Status Register
EE09	U9 Data Register
EE10	U8 (PIA-1) Output Register A
EE11	U8 Output Register B
EE12	U8 Control Register A
EE13	U8 Control Register B
EE20	U7 (PIA-2) Output Register A
EE21	U7 Output Register B
EE22	U7 Control Register A
EE23	U7 Control Register B
FC00 - FFFF	Read only Memory

Table 4.3 ACIA Serial Connector Signal Summary

PIN Number	Signal
1	+5V
2	On board Baud Clock
3	Gnd
4	Gnd
5	Gnd
6	+12V
7	-12V
8	$\overline{\text{Data Carrier Detect}}$
9	Output Data
10	$\overline{\text{Request to send}}$
11	Output clock
12	Input clock
13	Input data
14	$\overline{\text{Clear to send}}$

CHAPTER 5

SOFTWARE IMPLEMENTATION, RESULTS AND DISCUSSIONS

5.1 SOFTWARE IMPLEMENTATION

In this section, the implementation of composite highpass-integrating filter, Fast Fourier Transform and PSD measurement of track profile for 6800 microprocessor applications is described. A complete assembly program listing for these functions is given at Appencix 'A'. The memory requirement for the individual programs is given here. For higher flexibility the data memory locations for filter and PSD are considered separately for testing. To make up the complete track monitoring routine, these locations should be changed properly.

5.1.1 High Pass filter and Integration

This program takes approximately 1K of the memory (1146 locations) and is implemented in 32 bit organization for higher accuracy. This is assembled from Memory address 5000_{16} to 5477_{16}. The data buffer consists of four sections and is assembled from 2000_{16} to $27FF_{16}$ (Buffer 1), 2900_{16} to $30FF_{16}$ (Buffer 2), 3200_{16} to $39FF_{16}$ (Buffer 3), 4100_{16} to $48FF_{16}$ (Buffer 4), each a 2K section of the memory. The data stored in the buffer 1 is first filtered, integrated and then stored in buffer 3 and again the data stored in buffer 3 (first high pass and integration) is again processed to get the double

integrated version of the original data and is stored in buffer 1. Then the control goes over to Buffer 2 for processing the next section of the data. Buffer 2, Buffer 4 are used for this. After double integration of the data in buffer 2, the control again goes back to buffer 1 and this repeats.

5.1.2 Fast Fourier Transform

The FFT routine takes approximately 4.5 K of memory (4580 locations) including the data locations for real and imaginary parts each a 1K section (for 512 points) and is assembled from 4000_{16} to $51E4_{16}$. The data locations for real part start from $49CA_{16}$ to $4DC9_{16}$ and $4DCA_{16}$ to $51C9_{16}$ are for imaginary part. The data on which FFT analysis is to be done is first stored in these locations and after the run, the result is available at the same locations.

5.1.3 PSD Measurement

This routine takes 353 locations of the memory and uses the multiplication routine in the FFT program. This is assembled in two parts. One part is a high pass filter for detrending the data (removal of DC and very low frequencies) and the second part is for the PSD measurement. These are assembled from 5200_{16} to $52C2_{16}$ and 5500_{16} to $559C_{16}$.

5.2 RESULTS

Since the accelerometer data is not available, the high pass filter and integrator algorithm is tested for DC superimposed sine wave and the various outputs are given at Fig. 5.1 for a section of the data. This program is tested for various frequencies of the input data and found to be satisfactory. However, the PSD measurement through FFT is tested using a typical track profile data, supplied by RDSO (Research, Development and Systems Organization, Lucknow) and the results are shown in Fig. 5.2. This is also tested using standard FFT Fortran subroutines on DEC-1090 computer and the results are found to be satisfactory. Due to the limited precision of signal processor which uses the 6800 microprocessor, the accuracy is somewhat affected. The run time of double integration algorithm is approximately 2 seconds, whereas for FFT subroutine and PSD routine the time is approximately 18 seconds. This is mainly due to the software multiply routine which takes an approximate time of 1.67 m sec for each 16 x 16 BIT multiplication. A hardware multiplier using one of the TRW multiplier chips reduces the multiplication time by approximately 1/5, since the worst case timing including all the instructions and returns from the subroutine will be about 250 μ sec.

5.3 CONCLUSIONS

In this thesis, a complete hardware system for routine track monitoring is described in detail. All the individual hardware modules are tested and packed in a 10" x 12" x 12" box. All the software routines are tested. However, the complete system performance is not evaluated due to the lack of the appropriate data. But it is expected that the system works satisfactorily in the real time environment due to its consistent performance while testing individual components.

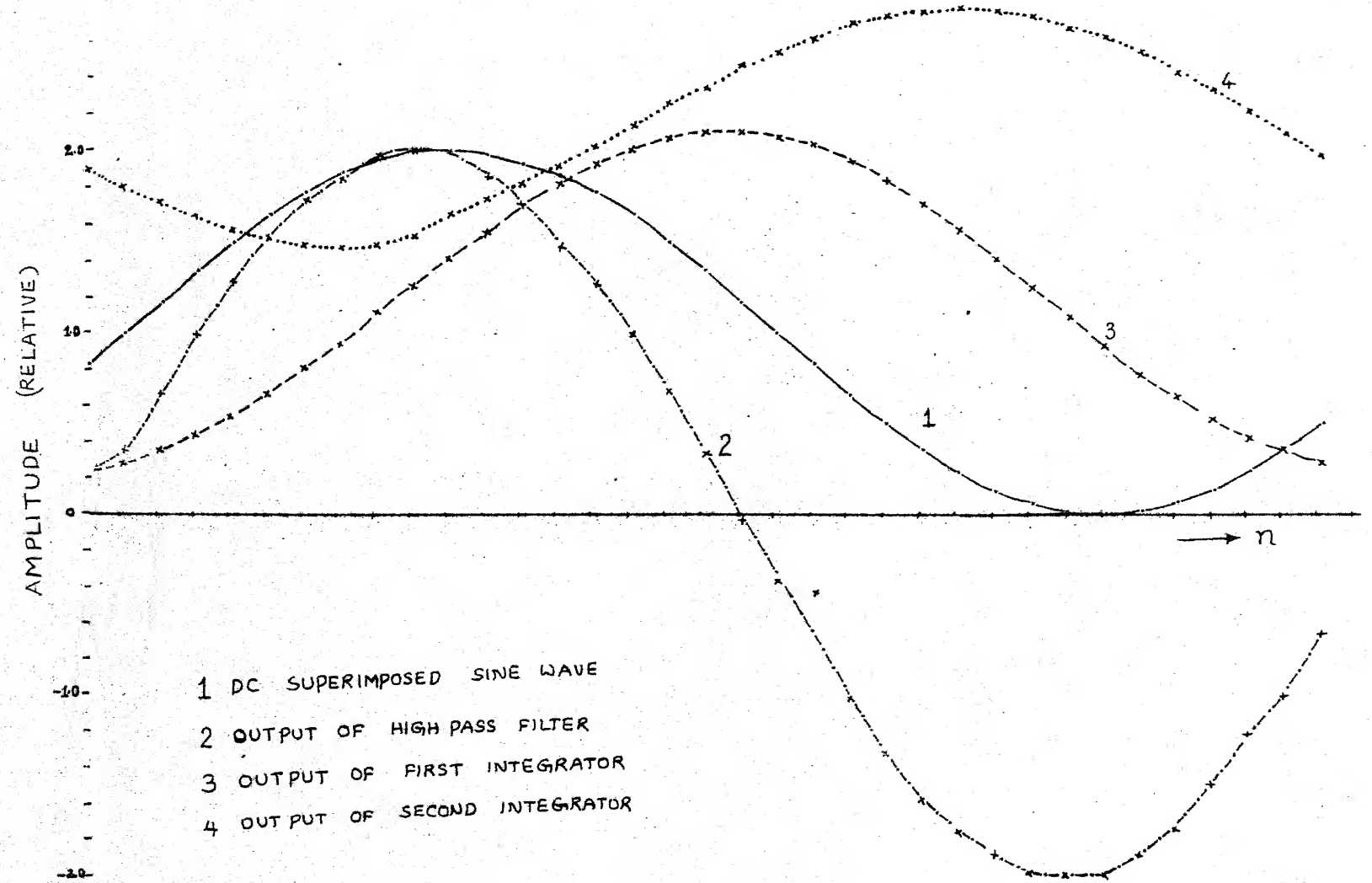
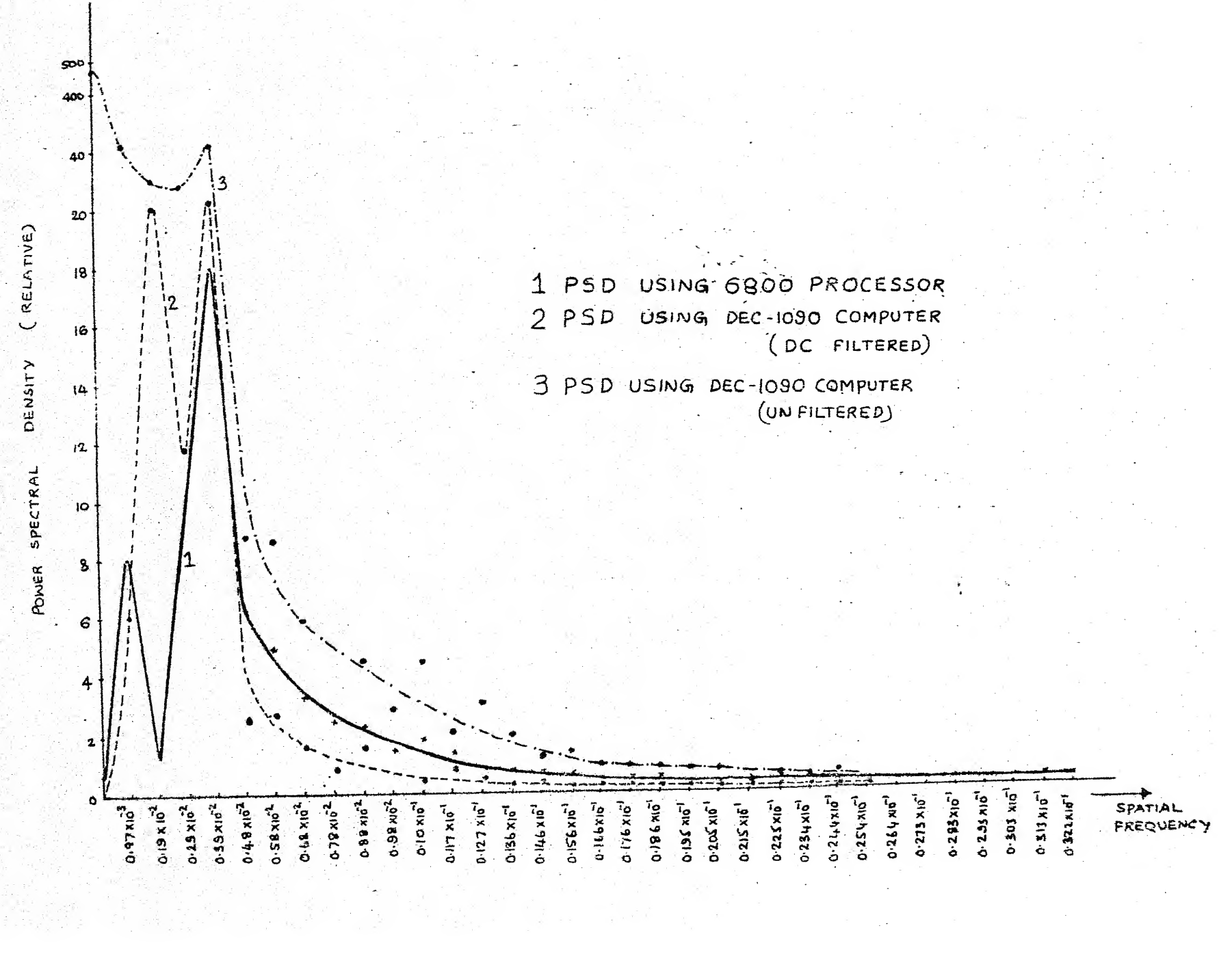


FIG. 5.1. TEST WAVE FORMS OF COMPOSITE FILTER


1 PSD USING 6800 PROCESSOR
2 PSD USING DEC-1090 COMPUTER (DC FILTERED)
3 PSD USING DEC-1090 COMPUTER (UN FILTERED)
POWER SPECTRAL DENSITY (RELATIVE)
SPATIAL FREQUENCY

LIST OF REFERENCES

1. L.R. Ranbiner and B. Gold, "Theory and Application of Digital Signal Processing", Prentice Hall India Ltd., 1979.

2. A. Antoniou, "Digital Filters: Analysis and Design", Tata McGraw-Hill Publishing Company Ltd., New Delhi, 1980.

3. K. Nandakumar, "Digital Filtering of Low Frequency Signals by the Method of Incremental Computation", M.Tech. Thesis, Department of Electrical Engineering, I.I.T. Kanpur, Jan. 1981.

4. K.L. Chug, "A FFT Program for Microcomputer for Real Time Application", M.Tech. Thesis, Department of Electrical Engineering, I.I.T. Kanpur, August, 1978.

5. Mischa Schwartz, Leonard Shaw, "Signal Processing: Discrete Spectral Analysis, Detection and Estimation", McGraw Hill Kogakusha, 1975.

M 6800

ASSEMBLY LISTING.

```
                        *       6800 ASSEMBLY PROGRAM
                        *       FOR FIRST ORDER INCRE-
                        *       MENTAL FILTER AND
                        *       SIMPSON INTEGRATOR
                        *       FOLLOWING ARE WORKING
                        *       LOCATIONS FOR THE PROGRAM
1FFC                    X1      EQU     $1FFC
28FC                    X2      EQU     $28FC
31FC                    Y1      EQU     $31FC
40FC                    Y2      EQU     $40FC
4905                    DELY    EQU     $4905
4910                    DELY0   EQU     $4910
4915                    DELY1   EQU     $4915
4920                    DELY2   EQU     $4920
4925                    DEL1    EQU     $4925
4930                    DEL2    EQU     $4930
4935                    INTER   EQU     $4935
4940                    INTER1  EQU     $4940
4945                    INTER2  EQU     $4945
4950                    TEMP    EQU     $4950
4955                    TEMP1   EQU     $4955
4960                    TEMP2   EQU     $4960
4975                    SHICOU  EQU     $4975
4980                    COUNT1  EQU     $4980
4985                    COUNT2  EQU     $4985
4990                    OUT0    EQU     $4990
4995                    OUT1    EQU     $4995
49A0                    OUT2    EQU     $49A0
                        *       MAIN PROGRAM START
5000                            ORG     $5000
5000  7F  1F FC                  CLR     X1
5003  7F  1F FD                  CLR     X1+1
5006  7F  1F FE                  CLR     X1+2
5009  7F  1F FF                  CLR     X1+3
500C  7F  28 FC                  CLR     X2
500F  7F  28 FD                  CLR     X2+1
5012  7F 28 FE                  CLR     X2+2
5015  7F 28 FF                  CLR     X2+3
5018  7F 31 FC                  CLR     Y1
501B  7F 31 FD                  CLR     Y1+1
501E  7F  31 FE                  CLR     Y1+2
5021  7F 31 FF                  CLR     Y1+3
5024  7F 40 FC                  CLR     Y2
5027  7F 40 FD                  CLR     Y2+1
502A  7F  40 FE                  CLR     Y2+2
502D  7F 40 FF                  CLR     Y2+3
5030  7F 49 05                  CLR     DELY
5033  7F  49 06                  CLR     DELY+1
5036  7F 49 07                  CLR     DELY+2
5039  7F 49 08                  CLR     DELY+3
503C  7F 49 10                  CLR     DELY0
503F  7F 49 11                  CLR     DELY0+1
5042  7F 49 12                  CLR     DELY0+2
5045  7F 49 13                  CLR     DELY0+3
5048  7F 49 15                  CLR     DELY1
```

```
504B  7F 49 16          CLR     DELY1+1
504E  7F 49 17          CLR     DELY1+2
5051  7F 49 18          CLR     DELY1+3
5054  7F 49 20          CLR     DELY2
5057  7F 49 21          CLR     DELY2+1
505A  7F 49 22          CLR     DELY2+2
505D  7F 49 23          CLR     DELY2+3
5060  7F 49 25          CLR     DEL1
5063  7F 49 26          CLR     DEL1+1
5066  7F 49 27          CLR     DEL1+2
5069  7F 49 28          CLR     DEL1+3
506C  7F 49 30          CLR     DEL2
506F  7F 49 31          CLR     DEL2+1
5072  7F 49 32          CLR     DEL2+2
5075  7F 49 33          CLR     DEL2+3
5078  7F 49 35          CLR     INTER
507B  7F 49 36          CLR     INTER+1
507E  7F 49 37          CLR     INTER+2
5081  7F 49 38          CLR     INTER+3
5084  7F 49 40          CLR     INTER1
5087  7F 49 41          CLR     INTER1+1
508A  7F 49 42          CLR     INTER1+2
508D  7F 49 43          CLR     INTER1+3
5090  7F 49 45          CLR     INTER2
5093  7F 49 46          CLR     INTER2+1
5096  7F 49 47          CLR     INTER2+2
5099  7F 49 48          CLR     INTER2+3
509C  7F 49 90          CLR     OUT0
509F  7F 49 91          CLR     OUT0+1
50A2  7F 49 92          CLR     OUT0+2
50A5  7F 49 93          CLR     OUT0+3
50A8  7F 49 95          CLR     OUT1
50AB  7F 49 96          CLR     OUT1+1
50AE  7F 49 97          CLR     OUT1+2
50B1  7F 49 98          CLR     OUT1+3
50B4  7F 49 A0          CLR     OUT2
50B7  7F 49 A1          CLR     OUT2+1
50BA  7F 49 A2          CLR     OUT2+2
50BD  7F 49 A3          CLR     OUT2+3
50C0  7F 49 50          CLR     TEMP
50C3  7F 49 55          CLR     TEMP1
50C6  7F 49 60          CLR     TEMP2
50C9  7F 49 75          CLR     SHICOU
50CC  CE 1F FF  OUTER   LDX     #$1FFF
50CF  FF 49 80          STX     COUNT1
50D2  CE 32 03          LDX     #$3203
50D5  FF 49 85          STX     COUNT2
                *
50D8  FE 49 80  MAIN    MAIN LOOP OF FILTER PROGRAM
50DB  8C 1F FF          LDX     COUNT1
50DE  27 12             CPX     #$1FFF
50E0  8C 28 FF          BEQ     INNER1
50E3  27 0D             CPX     #$28FF
50E5  8C 31 FF          BEQ     INNER1
50E8  27 6B             CPX     #$31FF
50EA  8C 40 FF          BEQ     INNER2
                        CPX     #$40FF
```

```
50ED  27 66          BEQ     INNER2
50EF  7E 51 B5       JMP     INNER
                     *       TAKING CARE OF DELAYED VALUES
                     *       FO THE FIRST LOOP
50F2  B6 49 43  INNER1 LDA A  INTER1+3
50F5  B7 49 38       STA A   INTER+3
50F8  B6 49 42       LDA A   INTER1+2
50FB  B7 49 37       STA A   INTER+2
50FE  B6 49 41       LDA A   INTER1+1
5101  B7 49 36       STA A   INTER+1
5104  B6 49 40       LDA A   INTER1
5107  B7 49 35       STA A   INTER
510A  B6 49 18       LDA A   DELY1+3
510D  B7 49 08       STA A   DELY+3
5110  B6 49 17       LDA A   DELY1+2
5113  B7 49 07       STA A   DELY+2
5116  B6 49 16       LDA A   DELY1+1
5119  B7 49 06       STA A   DELY+1
511C  B6 49 15       LDA A   DELY1
511F  B7 49 05       STA A   DELY
5122  B6 49 28       LDA A   DEL1+3
5125  B7 49 13       STA A   DELY0+3
5128  B6 49 27       LDA A   DEL1+2
512B  B7 49 12       STA A   DELY0+2
512E  B6 49 26       LDA A   DEL1+1
5131  B7 49 11       STA A   DELY0+1
5134  B6 49 25       LDA A   DEL1
5137  B7 49 10       STA A   DELY0
513A  B6 49 98       LDA A   OUT1+3
513D  B7 49 93       STA A   OUT0+3
5140  B6 49 97       LDA A   OUT1+2
5143  B7 49 92       STA A   OUT0+2
5146  B6 49 96       LDA A   OUT1+1
5149  B7 49 91       STA A   OUT0+1
514C  B6 49 95       LDA A   OUT1
514F  B7 49 90       STA A   OUT0
5152  7E 51 B5       JMP     INNER
                     *       TAKING CARE OF DELAYED VALUES
                     *       FOR THE SECOND LOOP
5155  B6 49 48  INNER2 LDA A  INTER2+3
5158  B7 49 38       STA A   INTER+3
515B  B6 49 47       LDA A   INTER2+2
515E  B7 49 37       STA A   INTER+2
5161  B6 49 46       LDA A   INTER2+1
5164  B7 49 36       STA A   INTER+1
5167  B6 49 45       LDA A   INTER2
516A  B7 49 35       STA A   INTER
516D  B6 49 23       LDA A   DELY2+3
5170  B7 49 08       STA A   DELY+3
5173  B6 49 22       LDA A   DELY2+2
5176  B7 49 07       STA A   DELY+2
5179  B6 49 21       LDA A   DELY2+1
517C  B7 49 06       STA A   DELY+1
517F  B6 49 20       LDA A   DELY2
5182  B7 49 05       STA A   DELY
5185  B6 49 33       LDA A   DEL2+3
```

```
5188  B7 49 13          STA A   DELY0+3
518B  B6 49 32          LDA A   DEL2+2
518E  B7 49 12          STA A   DELY0+2
5191  B6 49 31          LDA A   DEL2+1
5194  B7 49 11          STA A   DELY0+1
5197  B6 49 30          LDA A   DEL2
519A  B7 49 10          STA A   DELY0
519D  B6 49 A3          LDA A   OUT2+3
51A0  B7 49 93          STA A   OUT0+3
51A3  B6 49 A2          LDA A   OUT2+2
51A6  B7 49 92          STA A   OUT0+2
51A9  B6 49 A1          LDA A   OUT2+1
51AC  B7 49 91          STA A   OUT0+1
51AF  B6 49 A0          LDA A   OUT2
51B2  B7 49 90          STA A   OUT0
                    *           INNER LOOP OF THE PROGRAM
51B5  B6 49 38  INNER   LDA A   INTER+3
51B8  F6 49 37          LDA B   INTER+2
51BB  B0 49 08          SUB A   DELY+3
51BE  F2 49 07          SBC B   DELY+2
51C1  B7 49 38          STA A   INTER+3
51C4  B7 49 53          STA A   TEMP+3
51C7  F7 49 37          STA B   INTER+2
51CA  F7 49 52          STA B   TEMP+2
51CD  B6 49 36          LDA A   INTER+1
51D0  F6 49 35          LDA B   INTER
51D3  B2 49 06          SBC A   DELY+1
51D6  F2 49 05          SBC B   DELY
51D9  B7 49 36          STA A   INTER+1
51DC  B7 49 51          STA A   TEMP+1
51DF  F7 49 35          STA B   INTER
51E2  F7 49 50          STA B   TEMP
51E5  86 05             LDA A   #$05
51E7  B7 49 75          STA A   SHICOU
51EA  77 49 50  LOOP    ASR     TEMP
51ED  76 49 51          ROR     TEMP+1
51F0  76 49 52          ROR     TEMP+2
51F3  76 49 53          ROR     TEMP+3
51F6  7A 49 75          DEC     SHICOU
51F9  7D 49 75          TST     SHICOU
51FC  26 EC             BNE     LOOP
51FE  A6 04             LDA A   4,X
5200  E6 03             LDA B   3,X
5202  A0 00             SUB A   0,X
5204  09                DEX
5205  E2 00             SBC B   0,X
5207  B7 49 08          STA A   DELY+3
520A  F7 49 07          STA B   DELY+2
520D  09                DEX
520E  A6 04             LDA A   4,X
5210  E6 03             LDA B   3,X
5212  A2 00             SBC A   0,X
5214  09                DEX
5215  E2 00             SBC B   0,X
5217  08                INX
5218  08                INX
```

```
5219  08                    INX
521A  B7 49 06              STA A   DELY+1
521D  F7 49 05              STA B   DELY
5220  B6 49 08              LDA A   DELY+3
5223  F6 49 07              LDA B   DELY+2
5226  BB 49 53              ADD A   TEMP+3
5229  F9 49 52              ADC B   TEMP+2
522C  B7 49 08              STA A   DELY+3
522F  F7 49 07              STA B   DELY+2
5232  B6 49 06              LDA A   DELY+1
5235  F6 49 05              LDA B   DELY
5238  B9 49 51              ADC A   TEMP+1
523B  F9 49 50              ADC B   TEMP
523E  B7 49 06              STA A   DELY+1
5241  F7 49 05              STA B   DELY
5244  B6 49 08              LDA A   DELY+3
5247  F6 49 07              LDA B   DELY+2
524A  BB 49 13              ADD A   DELY0+3
524D  F9 49 12              ADC B   DELY0+2
5250  B7 49 13              STA A   DELY0+3
5253  B7 49 58              STA A   TEMP1+3
5256  F7 49 12              STA B   DELY0+2
5259  F7 49 57              STA B   TEMP1+2
525C  B6 49 06              LDA A   DELY+1
525F  F6 49 05              LDA B   DELY
5262  B9 49 11              ADC A   DELY0+1
5265  F9 49 10              ADC B   DELY0
5268  B7 49 11              STA A   DELY0+1
526B  B7 49 56              STA A   TEMP1+1
526E  F7 49 10              STA B   DELY0
5271  F7 49 55              STA B   TEMP1
5274  86 04                 LDA A   #$04
5276  B7 49 75              STA A   SHICOU
5279  77 49 55  LOOP1       ASR     TEMP1
527C  76 49 56              ROR     TEMP1+1
527F  76 49 57              ROR     TEMP1+2
5282  76 49 58              ROR     TEMP1+3
5285  7A 49 75              DEC     SHICOU
5288  7D 49 75              TST     SHICOU
528B  26 EC                 BNE     LOOP1
528D  FF 49 60              STX     TEMP2
                      *     BRANCHING TO INTEGRATION ROUTINE
5290  BD 54 46              JSR     INTEGR
5293  FE 49 85              LDX     COUNT2
5296  B6 49 58              LDA A   TEMP1+3
5299  A7 00                 STA A   0,X
529B  09                    DEX
529C  B6 49 57              LDA A   TEMP1+2
529F  A7 00                 STA A   0,X
52A1  09                    DEX
52A2  B6 49 56              LDA A   TEMP1+1
52A5  A7 00                 STA A   0,X
52A7  09                    DEX
52A8  B6 49 55              LDA A   TEMP1
52AB  A7 00                 STA A   0,X
52AD  FE 49 60              LDX     TEMP2
```

```
52B0  8C 27 FB          CPX     #$27FB
52B3  27 12             BEQ     STORE1
52B5  8C 30 FB          CPX     #$30FB
52B8  27 23             BEQ     STOR2
52BA  8C 39 FB          CPX     #$39FB
52BD  27 20             BEQ     STOR5
52BF  8C 48 FB          CPX     #$48FB
52C2  27 1D             BEQ     STOR6
52C4  7E 54 00          JMP     NEXT
                *       TAKING CARE OF LAST
                *       LOCATION VALUES
52C7  A6 04     STORE1  LDA A   4,X
52C9  B7 28 FF          STA A   X2+3
52CC  A6 03             LDA A   3,X
52CE  B7 28 FE          STA A   X2+2
52D1  A6 02             LDA A   2,X
52D3  B7 28 FD          STA A   X2+1
52D6  A6 01             LDA A   1,X
52D8  B7 28 FC          STA A   X2
52DB  20 06             BRA     PRST01
52DD  20 71     STOR2   BRA     STORE2
52DF  20 6A     STOR5   BRA     STORE5
52E1  20 6A     STOR6   BRA     STORE6
52E3  B6 49 08  PRST01  LDA A   DELY+3
52E6  B7 49 18          STA A   DELY1+3
52E9  B6 49 07          LDA A   DELY+2
52EC  B7 49 17          STA A   DELY1+2
52EF  B6 49 06          LDA A   DELY+1
52F2  B7 49 16          STA A   DELY1+1
52F5  B6 49 05          LDA A   DELY
52F8  B7 49 15          STA A   DELY1
52FB  B6 49 38          LDA A   INTER+3
52FE  B7 49 43          STA A   INTER1+3
5301  B6 49 37          LDA A   INTER+2
5304  B7 49 42          STA A   INTER1+2
5307  B6 49 36          LDA A   INTER+1
530A  B7 49 41          STA A   INTER1+1
530D  B6 49 35          LDA A   INTER
5310  B7 49 40          STA A   INTER1
5313  B6 49 13          LDA A   DELY0+3
5316  B7 49 28          STA A   DEL1+3
5319  B6 49 12          LDA A   DELY0+2
531C  B7 49 27          STA A   DEL1+2
531F  B6 49 11          LDA A   DELY0+1
5322  B7 49 26          STA A   DEL1+1
5325  B6 49 10          LDA A   DELY0
5328  B7 49 25          STA A   DEL1
532B  B6 49 93          LDA A   OUT0+3
532E  F6 49 92          LDA B   OUT0+2
5331  B7 49 98          STA A   OUT1+3
5334  F7 49 97          STA B   OUT1+2
5337  B6 49 91          LDA A   OUT0+1
533A  F6 49 90          LDA B   OUT0
533D  B7 49 96          STA A   OUT1+1
5340  F7 49 95          STA B   OUT1
5343  8C 30 FB          CPX     #$30FB
```

```
5346  27 1F          BEQ     FOLLO1
5348  7E 54 17       JMP     SECOND
534B  20 1D    STORE5  BRA     STORE3
534D  7E 53 E6 STORE6  JMP     STORE4
5350  A6 04    STORE2  LDA A   4,X
5352  B7 1F FF       STA A   X1+3
5355  A6 03          LDA A   3,X
5357  B7 1F FE       STA A   X1+2
535A  A6 02          LDA A   2,X
535C  B7 1F FD       STA A   X1+1
535F  A6 01          LDA A   1,X
5361  B7 1F FC       STA A   X1
5364  7E 52 E3       JMP     PRSTO1
5367  7E 54 26 FOLLO1  JMP     THIRD
536A  A6 04    STORE3  LDA A   4,X
536C  B7 40 FF       STA A   Y2+3
536F  A6 03          LDA A   3,X
5371  B7 40 FE       STA A   Y2+2
5374  A6 02          LDA A   2,X
5376  B7 40 FD       STA A   Y2+1
5379  A6 01          LDA A   1,X
537B  B7 40 FC       STA A   Y2
537E  B6 49 08 PRSTO2  LDA A   DELY+3
5381  B7 49 23       STA A   DELY2+3
5384  B6 49 07       LDA A   DELY+2
5387  B7 49 22       STA A   DELY2+2
538A  B6 49 06       LDA A     DELY+1
538D  B7 49 21       STA A   DELY2+1
5390  B6 49 05       LDA A   DELY
5393  B7 49 20       STA A   DELY2
5396  B6 49 38       LDA A   INTER+3
5399  B7 49 48       STA A   INTER2+3
539C  B6 49 37       LDA A   INTER+2
539F  B7 49 47       STA A   INTER2+2
53A2  B6 49 36       LDA A   INTER+1
53A5  B7 49 46       STA A   INTER2+1
53A8  B6 49 35       LDA A   INTER
53AB  B7 49 45       STA A   INTER2
53AE  B6 49 13       LDA A   DELY0+3
53B1  B7 49 33       STA A   DEL2+3
53B4  B6 49 12       LDA A   DELY0+2
53B7  B7 49 32       STA A   DEL2+2
53BA  B6 49 11       LDA A   DELY0+1
53BD  B7 49 31       STA A   DEL2+1
53C0  B6 49 10       LDA A   DELY0
53C3  B7 49 30       STA A   DEL2
53C6  B6 49 93       LDA A   OUT0+3
53C9  B7 49 A3       STA A   OUT2+3
53CC  B6 49 92       LDA A   OUT0+2
53CF  B7 49 A2       STA A   OUT2+2
53D2  B6 49 91       LDA A   OUT0+1
53D5  B7 49 A1       STA A   OUT2+1
53D8  B6 49 90       LDA A   OUT0
53DB  B7 49 A0       STA A   OUT2
53DE  8C 48 FB       CPX     #$48FB
53E1  27 1A          BEQ     FOLLO2
```

```
53E3  7E 54 35          JMP     FOURTH
53E6  A6 04       STORE4 LDA A  4,X
53E8  B7 31 FF          STA A   Y1+3
53EB  A6 03             LDA A   3,X
53ED  B7 31 FE          STA A   Y1+2
53F0  A6 02             LDA A   2,X
53F2  B7 31 FD          STA A   Y1+1
53F5  A6 01             LDA A   1,X
53F7  B7 31 FC          STA A   Y1
53FA  7E 53 7E          JMP     PRSTO2
53FD  7E 54 44    FOLLO2 JMP    OUT
5400  FE 49 80    NEXT   LDX    COUNT1
5403  08                 INX
5404  08                 INX
5405  08                 INX
5406  08                 INX
5407  FF 49 80           STX    COUNT1
540A  FE 49 85           LDX    COUNT2
540D  08                 INX
540E  08                 INX
540F  08                 INX
5410  08                 INX
5411  FF 49 85           STX    COUNT2
5414  7E 50 D8           JMP    MAIN
5417  CE 31 FF  SECOND   LDX    #$31FF
541A  FF 49 80           STX    COUNT1
541D  CE 20 03           LDX    #$2003
5420  FF 49 85           STX    COUNT2
5423  7E 50 D8           JMP    MAIN
5426  CE 40 FF  THIRD    LDX    #$40FF
5429  FF 49 80           STX    COUNT1
542C  CE 29 03           LDX    #$2903
542F  FF 49 85           STX    COUNT2
5432  7E 50 D8           JMP    MAIN
5435  CE 28 FF  FOURTH   LDX    #$28FF
5438  FF 49 80           STX    COUNT1
543B  CE 41 03           LDX    #$4103
543E  FF 49 85           STX    COUNT2
5441  7E 50 D8           JMP    MAIN
5444  3E           OUT   WAI
5445  39                 RTS
                         SUBROUTINE FOR INTEGRATION
5446  01         INTEGR  NOP
5447  B6 49 82           LDA A  TEMP1+3
544A  F6 49 81           LDA B  TEMP1+2
544D  BB 49 83           ADD A  OUT3+3
5450  F9 49 82           ADC B  OUT3+2
5453  B7 49 83           STA A  OUT3+3
5456  F7 49 82           STA B  OUT3+2
5459  B7 49 87           STA A  TEMP1+3
545C  F7 49 86           STA B  TEMP1+2
545F  26 49 85           LDA B  TEMP1+1
5462  86 49 84           LDA A  TEMP1
5465  F9 49 81           ADC B  OUT3+1
5468  B9 49 80           ADC A  OUT3
546B  B7 49 81           STA A  OUT3+1
```

```
546E  F7 49 90          STA B   OUTØ
5471  B7 49 56          STA A   TEMP1+1
5474  F7 49 55          STA B   TEMP1
5477  39                RTS
5478                    END

STATEMENTS =453

FREE BYTES =19460

NO ERRORS DETECTED
--
```

```
                    *       FFT IN PLACE PROGRAM
                    *       N IS NO.OF SAMPLES
                    *       N SHOULD BE POWER OF TWO
0080                        ORG     $80
                    *
                     *       WORKING LOCATIONS FOR
                    *       MULTIPLICATION
                    *       USING BOOTHS ALGORITHM
                     *
0080                Y       RMB     2
0082                XX      RMB     2
0084                U       RMB     4
0088                FF      RMB     1
4000                        ORG     $4000
4000  01            FFTIN   NOP
4001  7E 43 74              JMP     FFT
                    *       257 VALUES OF SINE ARE
                    *       STORED FROM 0 TO PIE/2
                    *       INSTEPS OF 2PIE/1024
4004  00 00         SIN     FDB     $0000
4006  00 C9                 FDB     $00C9
4008  01 92                 FDB     $0192
400A  02 5B                 FDB     $025B
400C  03 24                 FDB     $0324
400E  03 FD                 FDB     $03FD
4010  04 B6                 FDB     $04B6
4012  05 7F                 FDB     $057F
4014  06 48                 FDB     $0648
4016  07 11                 FDB     $0711
4018  07  D9                 FDB     $07D9
401A  08 A2                 FDB     $08A2
401C  09 6 B                 FDB     $096B
401E  0A 33                 FDB     $0A33
4020  0A FB                 FDB     $0AFB
4022  0B C4                 FDB     $0BC4
4024  0C 8C                 FDB     $0C8C
4026  0D 54                 FDB     $0D54
4028  0E 1C                 FDB     $0E1C
402A  0E E4                 FDB     $0EE4
402C  0F  AB                 FDB     $0FAB
402E  10 73                 FDB     $1073
4030  11 3A                 FDB     $113A
4032  12 01                 FDB     $1201
4034  12 C8                 FDB     $12C8
4036  13 8F                 FDB     $138F
4038  14 55                 FDB     $1455
403A  15 1C                 FDB     $151C
403C  15 E2                 FDB     $15E2
403E  16 A8                 FDB     $16A8
4040  17 6E                 FDB     $176E
4042  18 33                 FDB     $1833
4044  18 F9                 FDB     $18F9
4046  19 BE                 FDB     $19BE
4048  1A 83                 FDB     $1A83
404A  1B 47                 FDB     $1B47
404C  1C C0                 FDB     $1CC0
```

```
404E  1C D0          FDB     $1CD0
4050  1D 93          FDB     $1D93
4052  1E 57          FDB     $1E57
4054  1F 1A          FDB     $1F1A
4056  1F DD          FDB     $1FDD
4058  20 9F          FDB     $209F
405A  21 62          FDB     $2162
405C  22 24          FDB     $2224
405E  22 E5          FDB     $22E5
4060  23 A7          FDB     $23A7
4062  24 67          FDB     $2467
4064  25 28          FDB     $2528
4066  25 E8          FDB     $25E8
4068  26 A8          FDB     $26A8
406A  27 68          FDB     $2768
406C  28 27          FDB     $2827
406E  28 E5          FDB     $28E5
4070  29 A4          FDB     $29A4
4072  2A 62          FDB     $2A62
4074  2B 1F          FDB     $2B1F
4076  2B DC          FDB     $2BDC
4078  2C 99          FDB     $2C99
407A  2D 55          FDB     $2D55
407C  2E 11          FDB     $2E11
407E  2E CC          FDB     $2ECC
4080  2F 87          FDB     $2F87
4082  30 42          FDB     $3042
4084  30 FC          FDB     $30FC
4086  31 B5          FDB     $31B5
4088  32 6E          FDB     $326E
408A  33 27          FDB     $3327
408C  33 DF          FDB     $33DF
408E  34 97          FDB     $3497
4090  35 4E          FDB     $354E
4092  36 04          FDB     $3604
4094  36 BA          FDB     $36BA
4096  37 70          FDB     $3770
4098  38 25          FDB     $3825
409A  38 D9          FDB     $38D9
409C  39 8D          FDB     $398D
409E  3A 40          FDB     $3A40
40A0  3A F3          FDB     $3AF3
40A2  3B A5          FDB     $3BA5
40A4  3C 57          FDB     $3C57
40A6  3D 08          FDB     $3D08
40A8  3D B8          FDB     $3DB8
40AA  3E 68          FDB     $3E68
40AC  3F 17          FDB     $3F17
40AE  3F C6          FDB     $3FC6
40B0  40 74          FDB     $4074
40B2  41 21          FDB     $4121
40B4  41 CE          FDB     $41CE
40B6  42 7A          FDB     $427A
40B8  43 26          FDB     $4326
40BA  43 D1          FDB     $43D1
40BC  44 7B          FDB     $447B
```

```
40BE  45 24          FDB     $4524
40C0  45 CD          FDB     $45CD
40C2  46 75          FDB     $4675
40C4  47 1D          FDB     $471D
40C6  47 C4          FDB     $47C4
40C8  48 6A          FDB     $486A
40CA  49 0F          FDB     $490F
40CC  49 B4          FDB     $49B4
40CE  4A 58          FDB     $4A58
40D0  4A FB          FDB     $4AFB
40D2  4B 9E          FDB     $4B9E
40D4  4C 40          FDB     $4C40
40D6  4C E1          FDB     $4CE1
40D8  4D 81          FDB     $4D81
40DA  4E 21          FDB     $4E21
40DC  4E C0          FDB     $4EC0
40DE  4F 5E          FDB     $4F5E
40E0  4F FB          FDB     $4FFB
40E2  50 98          FDB     $5098
40E4  51 34          FDB     $5134
40E6  51 CF          FDB     $51CF
40E8  52 69          FDB     $5269
40EA  53 03          FDB     $5303
40EC  53 9B          FDB     $539B
40EE  54 33          FDB     $5433
40F0  54 CA          FDB     $54CA
40F2  55 60          FDB     $5560
40F4  55 F6          FDB     $55F6
40F6  56 8A          FDB     $568A
40F8  57 1E          FDB     $571E
40FA  57 B1          FDB     $57B1
40FC  58 43          FDB     $5843
40FE  58 D4          FDB     $58D4
4100  59 64          FDB     $5964
4102  59 F4          FDB     $59F4
4104  5A 82          FDB     $5A82
4106  5B 10          FDB     $5B10
4108  5B 9D          FDB     $5B9D
410A  5C 29          FDB     $5C29
410C  5C B4          FDB     $5CB4
410E  5D 3E          FDB     $5D3E
4110  5D C8          FDB     $5DC8
4112  5E 50          FDB     $5E50
4114  5E D7          FDB     $5ED7
4116  5F 5E          FDB     $5F5E
4118  5F E4          FDB     $5FE4
411A  60 68          FDB     $6068
411C  60 EC          FDB     $60EC
411E  61 6F          FDB     $616F
4120  61 F1          FDB     $61F1
4122  62 72          FDB     $6272
4124  62 F2          FDB     $62F2
4126  63 71          FDB     $6371
4128  63 EF          FDB     $63EF
412A  64 6C          FDB     $646C
412C  64 E9          FDB     $64E9
```

```
412E  65 6 4          FDB    $6564
4130  65 DE           FDB    $65DE
4132  66 57           FDB    $6657
4134  66 DØ           FDB    $66DØ
4136  67 47           FDB    $6747
4138  67 BD           FDB    $67BD
413A  68 32           FDB    $6832
413C  68 A7           FDB    $68A7
413E  69 1A           FDB    $691A
4140  69 8C           FDB    $698C
4142  69 FD           FDB    $69FD
4144  6A 6E           FDB    $6A6E
4146  6A DD           FDB    $6ADD
4148  6B 4B           FDB    $6B4B
414A  6B B8           FDB    $6BB8
414C  6C 24           FDB    $6C24
414E  6C 8F           FDB    $6C8F
4150  6C F9           FDB    $6CF9
4152  6D 62           FDB    $6D62
4154  6D CA           FDB    $6DCA
4156  6E 31           FDB    $6E31
4158  6E 97           FDB    $6E97
415A  6E FB           FDB    $6EFB
415C  6F 5F           FDB    $6F5F
415E  6F C2           FDB    $6FC2
4160  70 23           FDB    $7023
4162  70 83           FDB    $7083
4164  70 E3           FDB    $70E3
4166  71 41           FDB    $7141
4168  71 9E           FDB    $719E
416A  71 FA           FDB    $71FA
416C  72 55           FDB    $7255
416E  72 AF           FDB    $72AF
4170  73 08           FDB    $7308
4172  73 5F           FDB    $735F
4174  73 B6           FDB    $73B6
4176  74 0B           FDB    $740B
4178  74 60           FDB    $7460
417A  74 B3           FDB    $74B3
417C  75 05           FDB    $7505
417E  75 56           FDB    $7556
4180  75 A6           FDB    $75A6
4182  75 F 4          FDB    $75F4
4184  76 42           FDB    $7642
4186  76  8E          FDB    $768E
4188  76  D9          FDB    $76D9
418A  77 23           FDB    $7723
418C  77  6C          FDB    $776C
418E  77 B4           FDB    $77B4
4190  77 FB           FDB    $77FB
4192  78 40           FDB    $7840
4194  78 85           FDB    $7885
4196  78 C8           FDB    $78C8
4198  79 0A           FDB    $790A
419A  79 4A           FDB    $794A
419C  79 8A           FDB    $798A
```

```
420A  00 00     N       FDB     $0000
                *       IA INITIAL VALUE IS N/2
420C  00 00     IA      FDB     $0000
                *       LL STORES LOG2N
420E  00 00     LL      FDB     $0000
4210  00 00     HH      FDB     $0000  ;
                *       FOLLOWING PARAMETERS SET
                *       OFFSETS FOR CALCULATING
                *       SINE AND COSINE VALUES
4212  02 00     CT512   FDB     $0200
4214  00 00     S1024   FDB     $0000
4216  04 00     C1024   FDB     $0400
4218  00 00     SI512   FDB     $0000
421A  00 00     SN512   FDB     $0000
421C  FE 00     C512    FDB     $FE00
                *       INNER LOOP PARAMETERS
421E  00 01     IB      FDB     $0001  ;
4220  00 00     IC      FDB     $0000  ;
4222  00 00     ID      FDB     $0000  ;
4224  00 00     I2D     FDB     $0000  ;
4226  00 00     K       FDB     $0000  ;
                *       SINE IS TEMPORARY
                *       LOCATION TO SINE
4228  00 00     SINE    FDB     $0000  ;
                *       COSIN IS TEMPORARY
                *       LOCATION TO COSINE
422A  00 00     COSIN   FDB     $0000  ;
422C  00        COUNT   FCB     $00
                *       FLAGS FOR TESTING
422D  00        FLAG1   FCB     $00    ;
422E  00        FLAG2   FCB     $00
422F  00         FLAG    FCB     $00
4230  00 00     EL      FDB     $0000
4232  00 00     EH      FDB     $0000
4234  00 00     DL      FDB     $0000
4236  00 00     DH      FDB     $0000
                 *       REAL PART CURRENT
                *       ADDRESS STORE
4238  00 00     MIR     FDB     $0000  ;
                *       IMAG.PART CURENT
                *       ADDRESS STORE
423A  00 00     MII     FDB     $0000  ;
423C  00 00     NMR     FDB     $0000
423E  00 00     NMI     FDB     $0000
                 *       CURRENT REAL PART STORE
4240  00 00     XYRMI   FDB     $0000  ;
                *       CURRENT IMAG.PART STORE
4242  00 00     XYIMI   FDB     $0000  ;
                *       WORKING LOCATIONS TO
                *       STORE REAL&IMAG.PARTS
4244  00 00     AREAL   FDB     $0000  ;
4246  00 00     AIMAG   FDB     $0000  ;
4248  00 00     REAL1   FDB     $0000
424A  00 00     BREAL   FDB     $0000
424C  00 00     IMAG1   FDB     $0000
424E  00 00     BIMAG   FDB     $0000
```

```
420A  00 00     N       FDB     $0000
                *       IA INITIAL VALUE IS N/2
420C  00 00     IA      FDB     $0000
                *       LL STORES LOG2N
420E  00 00     LL      FDB     $0000
4210  00 00     HH      FDB     $0000  ;
                *       FOLLOWING PARAMETERS SET
                *       OFFSETS FOR CALCULATING
                *       SINE AND COSINE VALUES
4212  02 00     CT512   FDB     $0200
4214  00 00     S1024   FDB     $0000
4216  04 00     C1024   FDB     $0400
4218  00 00     SI512   FDB     $0000
421A  00 00     SN512   FDB     $0000
421C  FE 00     C512    FDB     $FE00
                *       INNER LOOP PARAMETERS
421E  00 01     IB      FDB     $0001  ;
4220  00 00     IC      FDB     $0000  ;
4222  00 00     ID      FDB     $0000  ;
4224  00 00     I2D     FDB     $0000  ;
4226  00 00     K       FDB     $0000  ;
                *       SINE IS TEMPORARY
                *       LOCATION TO SINE
4228  00 00     SINE    FDB     $0000  ;
                *       COSIN IS TEMPORARY
                *       LOCATION TO COSINE
422A  00 00     COSIN   FDB     $0000  ;
422C  00        COUNT   FCB     $00
                *       FLAGS FOR TESTING
422D  00        FLAG1   FCB     $00    ;
422E  00        FLAG2   FCB     $00
422F  00         FLAG    FCB     $00
4230  00 00     EL      FDB     $0000
4232  00 00     EH      FDB     $0000
4234  00 00     DL      FDB     $0000
4236  00 00     DH      FDB     $0000
                 *       REAL PART CURRENT
                *       ADDRESS STORE
4238  00 00     MIR     FDB     $0000  ;
                *       IMAG.PART CURENT
                *       ADDRESS STORE
423A  00 00     MII     FDB     $0000  ;
423C  00 00     NMR     FDB     $0000
423E  00 00     NMI     FDB     $0000
                 *       CURRENT REAL PART STORE
4240  00 00     XYRMI   FDB     $0000  ;
                *       CURRENT IMAG.PART STORE
4242  00 00     XYIMI   FDB     $0000  ;
                *       WORKING LOCATIONS TO
                *       STORE REAL&IMAG.PARTS
4244  00 00     AREAL   FDB     $0000  ;
4246  00 00     AIMAG   FDB     $0000  ;
4248  00 00     REAL1   FDB     $0000
424A  00 00     BREAL   FDB     $0000
424C  00 00     IMAG1   FDB     $0000
424E  00 00     BIMAG   FDB     $0000
```

```
4250  00          FLAG4   FCB     $00
4251  00          BFLAG   FCB     $00
4252  00           LL8     FCB     $00
                  *       TEMP ADDRESS LOCATIONS
                  *       OF BIT REVERSAL ROUTINE
4253  00 00       TREAL   FDB     $0000  ;
4255  00 00       TIMAG   FDB     $0000  ;
                  *       BITRN STORES BIT REVERSED#
4257  00 00       BITRN   FDB     $0000  ;
                  *       MAJOR WORK LOCATIONS
                  *       OF FFT PROGRAM
4259  00          H       FCB     $00    ;
425A  00          L       FCB     $00    ;
425B  00          D       FCB     $00    ;
425C  00          E       FCB     $00    ;
425D  00          B1      FCB     $00    ;
425E  00          C       FCB     $00    ;
                  *       DPMUL IS A SUBROUTINE TO
                  *       MULTIPY TWO'S COMPLIMENT
                  *       NO. USING BOOTHS ALGOITHM
                  *       MULTIPLIER AND MULTIPLICAND
                  *       SHOULD BE STORED AT DE,HL
                   *       WORK LOCATIONS AND THE
                  *       RESULT WILL BE AT HL
425F  01           DPMUL   NOP
4260  FE 42 5B            LDX     D
4263  DF 80                STX     Y
4265  FE  42 59            LDX     H
4268  DF  82               STX     XX
426A  CE 00 05            LDX     #5
426D  4F                   CLR A
426E  A7 83         LP1    STA A   U-1,X
4270  09                   DEX
4271  26 F B               BNE     LP1
4273  CE 00 10            LDX     #16
4276  96  81        LP2     LDA A   Y+1
4278  84 01                AND A   #1
427A  16                   TAB
427B  98 88                EOR A   FF
427D  27 1D                BEQ     SHIFT
427F  5D                   TST B
4280  27 0E                BEQ     ADD
4282  96 85                LDA A   U+1
4284  D6 84                LDA B   U
4286  90 83                SUB A   XX+1
4288  D2 82                SBC B   XX
428A  97 85                STA A   U+1
428C  D7 84                STA B   U
428E  20 0C                BRA     SHIFT
4290  96  85        ADD     LDA A   U+1
4292  D6  84                LDA B   U
4294  9B 83                ADD A   XX+1
4296  D9 82                ADC B   XX
4298  97 85                STA A   U+1
429A  D7 8 4               STA B   U
429C  7F  00 88   SHIFT    CLR     FF
```

```
429F  76  00 80          ROR     Y
42A2  76  00 81          ROR     Y+1
42A5  79 00 88           ROL     FF
42A8  77  00 84          ASR     U
42AB  76  00 85          ROR     U+1
42AE  76  00 86          ROR     U+2
42B1  76 00 87           ROR     U+3
42B4  09                 DEX
42B5  26 BF              BNE     LP2
42B7  78 00 86           ASL     U+2
42BA  79 00 85           ROL     U+1
42BD  79 00 84           ROL     U
42C0  96 84              LDA A   U
42C2  D6 85              LDA B   U+1
42C4  B7 42 59           STA A   H
42C7  F7 42 5A           STA B   L
42CA  39                 RTS
                *        SUBROUTINE COMPL FINDS
                *        TWO'S COMPLIMENT OF 16
                *        BIT NUMBER STORED AT HL
                *        RESULT AT HL
42CB  01        COMPL    NOP
42CC  4F                 CLR A
42CD  B6 42 5A           LDA A   L
42D0  F6 42 59           LDA B   H
42D3  53                 COM B
42D4  43                 COM A
42D5  8B 01              ADD A   #$01
42D7  C9 00              ADC B   #$00
42D9  B7 42 5A           STA A   L
42DC  F7 42 59           STA B   H
42DF  39                 RTS
                *        SUBROUTINE OFL DIVIDES
                *        16 BIT NUMBER STORED
                *        AT HL BY 2;RESULT AT HL.
42E0  01        OFL      NOP
42E1  86 00              LDA A   #$00
42E3  B7  42 50           STA A   FLAG4
42E6  B6 42 59           LDA A   H
42E9  84 80              AND A   #$80
42EB  27 08              BEQ     DVIDE
42ED  86 01              LDA A   #$01
42EF  B7 42 50           STA A   FLAG4
42F2  BD 42 CB           JSR     COMPL
42F5  01        DVIDE    NOP
42F6  0C                 CLC
42F7  76 42 59           ROR     H
42FA  76 42 5A           ROR     L
42FD  B6 42 50           LDA A   FLAG4
4300  81 01              CMP A   #$01
4302  2B 03              BMI     OVER
4304  BD 42 CB           JSR     COMPL
4307  39        OVER     RTS
                *
                *        SUBROUTINE BITR FINDS
                *        BIT REVERSED NO. OF A
```

```
                    *       NO. STORED AT HL
                    *       RESULT AT HL

4308  01            BITR    NOP
4309  FE 42 59              LDX     H
430C  FF 42 5B              STX     D
430F  B6  42 51              LDA A   BFLAG
4312  81 01                 CMP A   #$01
4314  2B 3A                 BMI     LLLE8
4316  B6 42 5C  LLGT8       LDA A   E
4319  C6 08                 LDA B   #$08
431B  CE 00 00              LDX     #$00
431E  FF 42 59              STX     H
4321  0C            BRLP3   CLC
4322  79 42 5A              ROL     L
4325  79 42 59              ROL     H
4328  46                    ROR A
4329  24 07                 BCC     BRLP4
432B  FE 42 59              LDX     H
432E  08                    INX
432F  FF 42 59              STX     H
4332  5A            BRLP4   DEC B
4333  26 EC                 BNE     BRLP3
4335  F6 42 52              LDA B   LL8
4338  B6 42 5B              LDA A   D
433B  0C            BRLP5   CLC
433C  79 42 5A              ROL     L
433F  79 42 59              ROL     H
4342  46                    ROR A
4343  24 07                 BCC     BRLP6
4345  FE 42 59              LDX     H
4348  08                    INX
4349  FF 42 59              STX     H
434C  5A            BRLP6   DEC B
434D  26 EC                 BNE     BRLP5
434F  39                    RTS
4350  01            LLLE8   NOP
4351  CE 42 0E              LDX     #LL
4354  B6 42 5C              LDA A   E
4357  E6 00                 LDA B   ,X
4359  CE 00 00              LDX     #$00
435C  FF 42 59              STX     H
435F  0C            BRLP1   CLC
4360  79 42 5A              ROL     L
4363  79 42 59              ROL     H
4366  46                    ROR A
4367  24 07                 BCC     BRLP2
4369  FE 42 59              LDX     H
436C  08                    INX
436D  FF 42 59              STX     H
4370  5A            BRLP2   DEC B
4371  26 EC                 BNE     BRLP1
4373  39                    RTS
                    *
                    *       MAIN FFT PROGRAM STARTS
                    *       N HAS #OF SAMPLE POINTS
```

```
4374  FE 42 0A  FFT     LDX     N       ;
4377  7F 42 59          CLR     H
437A  7F 42 5A          CLR     L
437D  7F 42 5B          CLR     D
4380  7F  42 5C          CLR     E
4383  7F 42 5E          CLR     C
4386  7F 42 5D          CLR     B1
4389  FF 42 59          STX     H
                *       TO CALCULATE LOG2N
438C  BD 51 CC          JSR     LOG2N   ;
438F  0C                CLC
4390  76 42 59          ROR     H
4393  76 42 5A          ROR     L
4396  FE 42 59          LDX     H
4399  FF 42 0C          STX     IA
439C  FF 42 06          STX     NI2
439F  F6 42 5E          LDA B   C
43A2  86 08             LDA A   #$08
43A4  11                CBA
                *       CHECKING LOG2N>8 OR <8
43A5  2B 06             BMI     GT8     ;
43A7  4F                CLR A
43A8  B7 42 51          STA A   BFLAG
43AB  20 0B             BRA     LT8
43AD  86 01       GT8   LDA A   #$01
43AF  B7 42 51          STA A   BFLAG
43B2  17                TBA
43B3  80 08             SUB A   #$08
43B5  B7 42 52          STA A   LL8
43B8  17          LT8   TBA
43B9  B7 42 0E          STA A   LL
43BC  80 02             SUB A   #$02
43BE  16                TAB
43BF  0C                CLC
43C0  86 01             LDA A   #$01
43C2  46                ROR A
43C3  46          LOOP  ROR A
43C4  5A                DEC B
43C5  26 FC             BNE     LOOP
43C7  B7 42 10          STA A   HH
                  *       CALCULATION OF DIFFERENT
                 *        OFFSETS TO GENERATE
                 *        SINE AND COSINE VALUES
43CA  CE 40 04          LDX     #SIN    ;
43CD  FF  42 5B          STX     D       ;
43D0  B6 42 16          LDA A   C1024   ;
43D3  F6  42 17          LDA B   C1024+1
43D6  FB 42 5C          ADD B   E
43D9  B9 42 5B          ADC A   D
43DC  B7  42 14          STA A   S1024
43DF  F7  42 15          STA B   S1024+1
43E2  CE 40 04          LDX     #SIN
43E5  FF  42 5B          STX     D
43E8  B6  42 1D          LDA A   C512+1
43EB  F6  42 1C          LDA B   C512
43EE  BB 42 5C          ADD A   E
```

```
43F1  F9 42 5B          ADC B   D
43F4  B7  42 19          STA A   SI512+1
43F7  F7  42 18          STA B   SI512
43FA  CE 40 04          LDX     #SIN
43FD  FF  42 5B          STX     D
4400  B6 42 13          LDA A   CT512+1
4403  F6  42 12          LDA B   CT512
4406  BB 42 5C          ADD A   E
4409  F9 42 5B          ADC B   D
440C  B7 42 1B          STA A   SN512+1
440F  F7 42 1A          STA B   SN512
4412  4F                CLR A
4413  B7  42 08          STA A   I
4416  CE 00 01          LDX     #$0001
4419  FF  42 1E          STX     IB
                *       OUTER LOOP
                *       PARAMETERS SETTING
441C  CE 00 00  OUTER   LDX     #$0000   ;
441F  FF  42 20          STX     IC       ;
4422  FF  42 26          STX     K
4425  FE  42 0C          LDX     IA
4428  FF  42 59          STX     H
442B  FF  42 22          STX     ID
442E  0C                CLC
442F  79 42 5A          ROL     L
4432  79 42 59          ROL     H
4435  FE  42 59          LDX     H
4438  FF  42 24          STX     I2D
                *        INNER LOOP PARAMETERS
                *       SETTING
443B  FE  42 0C  INNER   LDX     IA       ;
443E  FF  42 5B          STX     D        ;
4441  FE  42 20          LDX     IC
4444  FF  42 59          STX     H
                *       POWER OF W CALCULATED
4447  01        ICIA    NOP
4448  0C                CLC
4449  76  42 5B          ROR     D
444C  76  42 5C          ROR     E
444F  25 08              BCS     OICIA
4451  76  42 59          ROR     H
4454  76  42 5A          ROR     L
4457  20 EE              BRA     ICIA
4459  01        OICIA   NOP
445A  BD 43 08          JSR     BITR
445D  4F                CLR A
445E  B6  42 10          LDA A   HH
4461  46          ICLP    ROR A
4462  81 00              CMP A   #$00
4464  27 08              BEQ     FINIS
4466  78 42 5A          ASL     L
4469  79 42 59          ROL     H
446C  20 F 3             BRA     ICLP
446E  01        FINIS   NOP
                *        CHECK   POWER OF W>256
                *       OR <=256
```

```
446F  01         GL256   NOP
4470  B6 42 59           LDA A   H
4473  81 01              CMP A   #$01
4475  25 3A              BCS     NLT257
4477  81 01              CMP A   #$01
4479  26 07              BNE     GT256
447B  F6  42 5A  ACHEC    LDA B   L
447E  C1 01              CMP B   #$01
4480  25 2F              BCS     NLT257
                 *       VALUES OF SINE AND COSIN
                 *       FOR W>256
4482  01         GT256   NOP
4483  78 42 5A           ASL     L
4486  79 42 59           ROL     H
4489  FE  42 59           LDX     H
448C  FF  42 5B           STX     D
448F  4F                  CLR A
4490  B6 42 15           LDA A   S1024+1
4493  F6  42 14           LDA B   S1024
4496  B0 42 5C           SUB A   E
4499  F2 42 5B           SBC B   D
449C  B7 42 5A           STA A   L
449F  F7  42 59           STA B   H
44A2  FE 42 59           LDX     H
44A5  A6 00              LDA A   ,X
44A7  08                 INX
44A8  E6  00              LDA B   ,X
44AA  CE 42 28           LDX     #SINE
44AD  A7 00              STA A   ,X
44AF  20 02              BRA     NEXT
44B1  20 2F      NLT257  BRA     LT257
44B3  08         NEXT    INX
44B4  E7  00              STA B   ,X
44B6  B6  42 19           LDA A   SI512+1
44B9  F6  42 18           LDA B   SI512
44BC  BB 42 5C           ADD A   E
44BF  F9 42 5B           ADC B   D
44C2  B7  42 5A           STA A   L
44C5  F7  42 59           STA B   H
44C8  4F                  CLR A
44C9  FE  42 59           LDX     H
44CC  A6 00              LDA A   ,X
44CE  08                 INX
44CF  E6 0 0              LDA B   ,X
44D1  CE 42 2B           LDX     #COSIN+1
44D4  43                 COM A
44D5  53                 COM B
44D6  CB 01              ADD B   #$01
44D8  89 00              ADC A   #$00
44DA  E7  00              STA B   ,X
44DC  09                 DEX
44DD  A7 00              STA A   ,X
44DF  7E 45 40           JMP     IN1
                  *       VALUES OF SINE AND COSIN
                 *       FOR W<=256
44E2  01         LT257   NOP
```

```
44E3  78 42 5A          ASL     L
44E6  79 42 59          ROL     H
44E9  FE  42 59          LDX     H
44EC  FF  42 5B          STX     D
44EF  CE 40 04          LDX     #SIN
44F2  FF  42 59          STX     H
44F5  B6  42 5A          LDA A   L
44F8  F6  42 59          LDA B   H
44FB  BB 42 5C          ADD A   E
44FE  F9 42 5B          ADC B   D
4501  B7  42 5A          STA A   L
4504  F7  42 59          STA B   H
4507  FE 42 59          LDX     H
450A  A6 00             LDA A   ,X
450C  08                INX
450D  E6  00             LDA B   ,X
450F  FF  42 59          STX     H
4512  CE 42 28          LDX     #SINE
4515  A7  00             STA A   ,X
4517  08                INX
4518  E7 00             STA B   ,X
451A  4F                CLR A
451B  B6 42 1B          LDA A   SN512+1
451E  F6  42 1A          LDA B   SN512
4521  B0 42 5C          SUB A   E
4524  F2 42 5B          SBC B   D
4527  B7  42 5A          STA A   L
452A  F7  42 59          STA B   H
452D  FE 42 59          LDX     H
4530  A6 00             LDA A   ,X
4532  08                INX
4533  E6  00             LDA B   ,X
4535  FF  42 59          STX     H
4538  CE 42 2A          LDX     #COSIN
453B  A7  00             STA A   ,X
453D  08                INX
453E  E7  00             STA B   ,X
4540  01          IN1   NOP
4541  FE 42 0C          LDX     IA
4544  FF  42 59          STX     H
4547  78  42 5A          ASL     L
454A  79 42 59          ROL     H
454D  FE  42 59          LDX     H
4550  FF  42 5B          STX     D
4553  CE 49 CA          LDX     #XREAL
4556  FF  42 59          STX     H
4559  B6 42 5A          LDA A   L
455C  F6  42 59          LDA B   H
455F  BB 42 5C          ADD A   E
4562  F9 42 5B          ADC B   D
4565  B7 42 39          STA A   MIR+1
4568  F7  42 38          STA B   MIR
456B  CE 4D CA          LDX     #XIMAG
456E  FF  42 59          STX     H
4571  B6  42 5A          LDA A   L
4574  F6  42 59          LDA B   H
```

```
4577  BB 42 5C          ADD A   E
457A  F9 42 5B          ADC B   D
457D  B7  42 3B          STA A   MII+1
4580  F7  42 3A          STA B   MII
4583  FE 42 20          LDX     IC
4586  FF  42 59          STX     H
4589  78 42 5A          ASL     L
458C  79 42 59          ROL     H
458F  B6 42 5A          LDA A   L
4592  F6  42 59          LDA B   H
4595  B7  42 5E          STA A   C
4598  F7  42 5D          STA B   B1
                *         INNER MOST LOOP
459B  01          IMOST   NOP
459C  B6  42 5E          LDA A   C
459F  F6  42 5D          LDA B   B1
45A2  36                 PSH A
45A3  37                 PSH B
45A4  BB 42 39          ADD A   MIR+1
45A7  F9 42 38          ADC B   MIR
45AA  B7  42 5A          STA A   L
45AD  F7  42 59          STA B   H
45B0  FE  42 59          LDX     H
45B3  A6  00             LDA A   ,X
45B5  08                 INX
45B6  E6  00             LDA B   ,X
45B8  FF  42 5B          STX     D
45BB  B7  42 59          STA A   H
45BE  F7  42 5A          STA B   L
45C1  BD 42 E0          JSR     OFL
45C4  FE  42 59          LDX     H
45C7  FF  42 40          STX     XYRMI
45CA  FF  42 5B          STX     D
45CD  FE 42 2A          LDX     COSIN
45D0  FF  42 59          STX     H
45D3  BD 42 5F          JSR     DPMUL
45D6  FE 42 59          LDX     H
45D9  FF  42 48          STX     REAL1
45DC  33                 PUL B
45DD  32                 PUL A
45DE  BB 42 3B          ADD A   MII+1
45E1  F9 42 3A          ADC B   MII
45E4  B7  42 5A          STA A   L
45E7  F7  42 59          STA B   H
45EA  FE  42 59          LDX     H
45ED  A6  00             LDA A   ,X
45EF  08                 INX
45F0  E6  00             LDA B   ,X
45F2  FF  42 5B          STX     D
45F5  B7  42 59          STA A   H
45F8  F7  42 5A          STA B   L
45FB  BD 42 E0          JSR     OFL
45FE  FE 42 59          LDX     H
4601  FF  42 42          STX     XYIMI
4604  FF  42 5B          STX     D
4607  FE 42 28          LDX     SINE
```

```
460A  FF  42 59        STX     H
460D  B6  42 5E        LDA A   C
4610  F6  42 5D        LDA B   B1
4613  36               PSH A
4614  37               PSH B
4615  BD 42 5F         JSR     DPMUL
4618  FE 42 59         LDX     H
461B  FF  42 5B        STX     D
461E  B6 42 49         LDA A   REAL1+1
4621  F6  42 48        LDA B   REAL1
4624  BB 42 5C         ADD A   E
4627  F9 42 5B         ADC B   D
462A  B7  42 4B        STA A   BREAL+1
462D  F7  42 4A        STA B   BREAL
4630  FE  42 42        LDX     XYIMI
4633  FF  42 5B        STX     D
4636  FE  42 2A        LDX     COSIN
4639  FF  42 59        STX     H
463C  BD 42 5F         JSR     DPMUL
463F  FE 42 59         LDX     H
4642  FF  42 4C        STX     IMAG1
4645  FE  42 40        LDX     XYRMI
4648  F F 42 5B        STX     D
464B  FE  42 28        LDX     SINE
464E  FF  42 59        STX     H
4651  BD 42 5F         JSR     DPMUL
4654  FE 42 59         LDX     H
4657  BD 42 CB         JSR     COMPL
465A  B6  42 5A        LDA A   L
465D  F6  42 59        LDA B   H
4660  BB 42 4D         ADD A   IMAG1+1
4663  F9 42 4C         ADC B   IMAG1
4666  B7  42 5A        STA A   L
4669  F7  42 59        STA B   H
466C  FE 42 59         LDX     H
466F  FF  42 4E        STX     BIMAG
4672  33               PUL B
4673  32               PUL A
4674  B7 42 5E         STA A   C
4677  F7  42 5D        STA B   B1
467A  CE 49 CA         LDX     #XREAL
467D  FF  42 59        STX     H
4680  BB 42 5A         ADD A   L
4683  F9 42 59         ADC B   H
4686  B7  42 5A        STA A   L
4689  F7  42 59        STA B   H
468C  FE 42 59         LDX     H
468F  A6 0 0           LDA A   ,X
4691  08               INX
4692  E6  00           LDA B   ,X
4694  B7 42 59         STA A   H
4697  F7  42 5A        STA B   L
469A  BD 42 E0         JSR     OFL
469D  FE 42 59         LDX     H
46A0  FF 42 44         STX     AREAL
46A3  CE 4D CA         LDX     #XIMAG
```

```
46A6  FF  42 59          STX     H
46A9  B6  42 5E          LDA A   C
46AC  F6  42 5D          LDA B   B1
46AF  BB 42 5A          ADD A   L
46B2  F9 42 59          ADC B   H
46B5  B7  42 5A          STA A   L
46B8  F7  42 59          STA B   H
46BB  FE  42 59          LDX     H
46BE  A6 ØØ             LDA A   ,X
46CØ  Ø8                INX
46C1  E6  ØØ             LDA B   ,X
46C3  B7  42 59          STA A   H
46C6  F7  42 5A          STA B   L
46C9  BD 42 EØ          JSR     OFL
46CC  FE  42 59          LDX     H
46CF  FF  42 46          STX     AIMAG
46D2  B6  42 4B          LDA A   BREAL+1
46D5  F6  42 4A          LDA B   BREAL
46D8  BB 42 45          ADD A   AREAL+1
46DB  F9 42 44          ADC B   AREAL
46DE  B7 42 5C          STA A   E
46E1  F7  42 5B          STA B   D
46E4  CE 49 CA          LDX     #XREAL
46E7  FF  42 59          STX     H
46EA  B6 42 5A          LDA A   L
46ED  F6  42 59          LDA B   H
46FØ  BB 42 5E          ADD A   C
46F3  F9 42 5D          ADC B   B1
46F6  B7  42 5A          STA A   L
46F9  F7  42 59          STA B   H
46FC  FE 42 59          LDX     H
46FF  B6 42 5B          LDA A   D
4702  F6  42 5C          LDA B   E
4705  A7 ØØ             STA A   ,X
4707  Ø8                INX
4708  E7  ØØ             STA B   ,X
                    *     A REAL-B REAL CALCULATED
470A  FE  42 4A          LDX     BREAL
470D  FF  42 59          STX     H
4710  BD 42 CB          JSR     COMPL
4713  B6  42 5A          LDA A   L
4716  F6 42 59          LDA B   H
4719  BB 42 45          ADD A   AREAL+1
471C  F9 42 44          ADC B   AREAL
471F  B7 42 5C          STA A   E
4722  F7  42 5B          STA B   D
4725  B6 42 5E          LDA A   C
4728  F6  42 5D          LDA B   B1
472B  BB 42 39          ADD A   MIR+1
472E  F9 42 38          ADC B   MIR
4731  B7  42 5A          STA A   L
4734  F7  42 59          STA B   H
4737  FE 42 59          LDX     H
473A  B6  42 5B          LDA A   D
473D  A7  ØØ             STA A   ,X
473F  Ø8                INX
```

```
4740  F6  42 5C          LDA B   E
4743  E7  00             STA B   ,X
                  *      AIMAG+BIMAG CALCULATED
4745  B6  42 4F          LDA A   BIMAG+1
4748  F6  42 4E          LDA B   BIMAG
474B  BB 42 47           ADD A   AIMAG+1
474E  F9 42 46           ADC B   AIMAG
4751  B7  42 5C          STA A   E
4754  F7  42 5B          STA B   D
4757  CE 4D CA           LDX     #XIMAG
475A  FF  42 59          STX     H
475D  B6  42 5A          LDA A   L
4760  F6  42 59          LDA B   H
4763  BB 42 5E           ADD A   C
4766  F9 42 5D           ADC B   B1
4769  B7 42 5A           STA A   L
476C  F7  42 59          STA B   H
476F  FE 42 59           LDX     H
4772  B6  42 5B          LDA A   D
4775  A7 00              STA A   ,X
4777  08                 INX
4778  F6  42 5C          LDA B   E
477B  E7  00             STA B   ,X
477D  FE  42 4E          LDX     BIMAG
4780  FF  42 59          STX     H
4783  BD 42 CB           JSR     COMPL
4786  FE  42 59          LDX     H
4789  FF  42 5B          STX     D
478C  B6  42 47          LDA A   AIMAG+1
478F  F6  42 46          LDA B   AIMAG
4792  BB 42 5C           ADD A   E
4795  F9 42 5B           ADC B   D
4798  B7  42 5C          STA A   E
479B  F7  42 5B          STA B   D
479E  B6 42 3B           LDA A   MII+1
47A1  F6  42 3A          LDA B   MII
47A4  BB 42 5E           ADD A   C
47A7  F9 42 5D           ADC B   B1
47AA  B7 42 5A           STA A   L
47AD  F7  42 59          STA B   H
47B0  FE  42 59          LDX     H
47B3  B6  42 5B          LDA A   D
47B6  A7  00               STA A   ,X
47B8  08                 INX
47B9  F6  42 5C          LDA B   E
47BC  E7  00             STA B   ,X
47BE  FE  42 5D          LDX     B1
47C1  08                 INX
47C2  08                 INX
47C3  FF  42 5D          STX     B1
47C6  B6  42 5E          LDA A   C
47C9  CE 42 25           LDX     #I2D+1
47CC  A1 00              CMP A   ,X
47CE  26  02             BNE     IMOST1
47D0  20 03              BRA     NEXT3
                         NEXT ITERATION OF
```

```
                        *       INNER MOST LOOP
47D2  7E 45 9B  IMOST1  JMP     IMOST     ;
47D5  09        NEXT3   DEX               ;
47D6  B6 42 5D          LDA A   B1        ;
47D9  A1 00             CMP A   ,X
47DB  26 02             BNE     IMOST2
47DD  20 03             BRA     NEXT4
47DF  7E 45 9B  IMOST2  JMP     IMOST
                        *       NEXT ITERATION OF
                        *       INNER LOOP
47E2  FE 42 0C  NEXT4   LDX     IA        ;
47E5  FF 42 59          STX     H         ;
47E8  78 42 5A          ASL     L         ;
47EB  79 42 59          ROL     H
47EE  FE 42 59          LDX     H
47F1  FF 42 5B          STX     D
47F4  B6 42 21          LDA A   IC+1
47F7  F6  42 20          LDA B   IC
47FA  BB 42 5C          ADD A   E
47FD  F9 42 5B          ADC B   D
4800  B7 42 21          STA A   IC+1
4803  F7  42 20          STA B   IC
4806  B6 42 23          LDA A   ID+1
4809  F6  42 22          LDA B   ID
480C  BB 42 5C          ADD A   E
480F  F9 42 5B          ADC B   D
4812  B7  42 23          STA A   ID+1
4815  F7  42 22          STA B   ID
4818  48                ASL A
4819  59                ROL B
481A  B7 42 25          STA A   I2D+1
481D  F7  42 24          STA B   I2D
4820  FE 42 26          LDX     K
4823  08                INX
4824  FF  42 26          STX     K
4827  FF  42 5B          STX     D
482A  CE 42 1F          LDX     #IB+1
482D  B6 42 5C          LDA A   E
4830  A1 00             CMP A   ,X
4832  26 02             BNE     INNER1
4834  20 03             BRA     NEXT5
4836  7E  44 3B  INNER1  JMP     INNER
4839  09        NEXT5   DEX
483A  B6 42 5B          LDA A   D
483D  A1 00             CMP A   ,X
483F  26 02             BNE     INNER2
4841  20 03             BRA     NEXT6
4843  7E 44 3B  INNER2  JMP     INNER
                        *       NEXT ITERATION OF
                        *       OUTER LOOP
4846  4F        NEXT6   CLR A
4847  B6 42 1F          LDA A   IB+1
484A  F6  42 1E          LDA B   IB
484D  49                ROL A
484E  59                ROL B
484F  B7 42 1F          STA A   IB+1
```

```
4852  F7 42 1E          STA B   IB
4855  4F                CLR A
4856  B6 42 ØD          LDA A   IA+1
4859  F6 42 ØC          LDA B   IA
485C  56                ROR B
485D  46                ROR A
485E  B7 42 ØD          STA A   IA+1
4861  F7 42 ØC          STA B   IA
4864  7C 42 Ø8          INC     I
4867  B6 42 Ø8          LDA A   I
486A  CE 42 ØE          LDX     #LL
486D  A1 ØØ             CMP A   ,X
486F  26 Ø2             BNE     OUTER1
4871  2Ø Ø3             BRA     NEXTE
4873  7E 44 1C  OUTER1  JMP     OUTER
                *       UNSCRAMBLING OF OUTPUT
4876  CE ØØ ØØ  NEXTE   LDX     #$ØØØØ
4879  FF 42 5D          STX     B1
487C  B6 42 5E  XLP1    LDA A   C
487F  F6 42 5D          LDA B   B1
4882  B7 42 5A          STA A   L
4885  F7 42 59          STA B   H
4888  36                PSH A
4889  37                PSH B
488A  BD 43 Ø8          JSR     BITR
488D  B6 42 59          LDA A   H
4890  B1 42 5D          CMP A   B1
4893  2B Ø4             BMI     NLPCON
4895  27 Ø5             BEQ     NCHECA
4897  2Ø Ø6             BRA     EXCHG
4899  7E 49 9E  NLPCON  JMP     LPCON
489C  7E 49 91  NCHECA  JMP     CHECA
489F  Ø1        EXCHG   NOP
48AØ  78 42 5A          ASL     L
48A3  79 42 59          ROL     H
48A6  FE 42 59          LDX     H
48A9  FF 42 57          STX     BITRN
48AC  FF 42 5B          STX     D
48AF  78 42 5E          ASL     C
48B2  79 42 5D          ROL     B1
48B5  CE 49 CA          LDX     #XREAL
48B8  FF 42 59          STX     H
48BB  B6 42 5A          LDA A   L
48BE  F6 42 59          LDA B   H
48C1  BB 42 5C          ADD A   E
48C4  F9 42 5B          ADC B   D
48C7  B7 42 5A          STA A   L
48CA  F7 42 59          STA B   H
48CD  FE 42 59          LDX     H
48DØ  A6 ØØ             LDA A   ,X
48D2  Ø8                INX
48D3  E6 ØØ             LDA B   ,X
48D5  Ø9                DEX
48D6  FF 42 5B          STX     D
48D9  B7 42 54          STA A   TREAL+1
48DC  F7 42 53          STA B   TREAL
```

```
48DF  CE 49 CA        LDX     #XREAL
48E2  FF  42 59        STX     H
48E5  B6 42 5A        LDA A   L
48E8  F6  42 59        LDA B   H
48EB  BB 42 5E        ADD A   C
48EE  F9 42 5D        ADC B   B1
48F1  B7 42 5A        STA A   L
48F4  F7 42 59        STA B   H
48F7  FE 42 59        LDX     H
48FA  A6 00           LDA A   ,X
48FC  08              INX
48FD  E6 00           LDA B   ,X
48FF  09              DEX
4900  FF 51 CA        STX     TEMP
4903  FE 42 5B        LDX     D
4906  A7 00           STA A   ,X
4908  08              INX
4909  E7 00           STA B   ,X
490B  FE 42 53        LDX     TREAL
490E  FF 42 59        STX     H
4911  FE 51 CA        LDX     TEMP
4914  B6 42 5A        LDA A   L
4917  A7 00           STA A   ,X
4919  08              INX
491A  B6 42 59        LDA A   H
491D  A7 00           STA A   ,X
491F  FE 42 57        LDX     BITRN
4922  FF 42 5B        STX     D
4925  CE 4D CA        LDX     #XIMAG
4928  FF  42 59        STX     H
492B  B6 42 5A        LDA A   L
492E  F6 42 59        LDA B   H
4931  BB 42 5C        ADD A   E
4934  F9 42 5B        ADC B   D
4937  B7 42 5A        STA A   L
493A  F7 42 59        STA B   H
493D  FE 42 59        LDX     H
4940  A6 00           LDA A   ,X
4942  08              INX
4943  E6 00           LDA B   ,X
4945  09              DEX
4946  FF 42 5B        STX     D
4949  B7 42 56        STA A   TIMAG+1
494C  F7  42 55        STA B   TIMAG
494F  CE 4D CA        LDX     #XIMAG
4952  FF  42 59        STX     H
4955  B6 42 5E        LDA A   C
4958  F6  42 5D        LDA B   B1
495B  BB 42 5A        ADD A   L
495E  F9 42 59        ADC B   H
4961  B7 42 5A        STA A   L
4964  F7 42 59        STA B   H
4967  FE 42 59        LDX     H
496A  A6 00           LDA A   ,X
496C  08              INX
496D  E6 00           LDA B   ,X
```

```
496F  09              DEX
4970  FF 51 CA        STX     TEMP
4973  FE 42 5B        LDX     D
4976  A7 00           STA A   ,X
4978  08              INX
4979  E7 00           STA B   ,X
497B  FE 42 55        LDX     TIMAG
497E  FF 42 59        STX     H
4981  FE 51 CA        LDX     TEMP
4984  B6 42 5A        LDA A   L
4987  A7 00           STA A   ,X
4989  B6 42 59        LDA A   H
498C  08              INX
498D  A7 00           STA A   ,X
498F  20 0D           BRA     LPCON
4991  B6 42 5A  CHECA  LDA A   L
4994  B1 42 5E        CMP A   C
4997  27 05           BEQ     LPCON
4999  2B 03           BMI     LPCON
499B  7E 48 9F        JMP     EXCHG
499E  33        LPCON  PUL B
499F  32               PUL A
49A0  8B 01            ADD A   #$01
49A2  25 02            BCS     NEXT8
49A4  20 01            BRA     NEXT7
49A6  5C        NEXT8  INC B
49A7  B7 42 5E  NEXT7  STA A   C
49AA  F7 42 5D         STA B   B1
49AD  FE 42 0A         LDX     N
49B0  FF 42 59         STX     H
49B3  B6 42 59         LDA A   H
49B6  B1 42 5D         CMP A   B1
49B9  27 03            BEQ     SCHEC
49BB  7E 48 7C         JMP     XLP1
49BE  B6 42 5A  SCHEC  LDA A   L
49C1  B1 42 5E         CMP A   C
49C4  27 03            BEQ     XEND
49C6  7E 48 7C         JMP     XLP1
                *      END OF FFT SUBROUTINE
49C9  39        XEND   RTS             ;
                *      XREAL STORES 512 REAL VALUES
49CA            XREAL  RMB     1024    ;
                *      XIMAG STORES 512 IMAG.VALUES
4DCA            XIMAG  RMB     1024    ;
                *      TEMP WORK LOCATIONS
51CA  00        TEMP   FCB     00      ;
51CB  00        TEMP1  FCB     00
                *
                *      LOG2N SUBROUTINE FINDS LOG.OF N
                *
51CC  C6 08     LOG2N  LDA B   #$08
51CE  B6 42 5A         LDA A   L
51D1  0C               CLC
51D2  46        POWER  ROR A
51D3  25 0D            BCS     OUT
51D5  7C 42 5E         INC     C
```

```
51D8  5A                DEC B
51D9  26 F7             BNE      POWER
51DB  C6 Ø8             LDA B    #$Ø8
51DD  B6 42 59          LDA A    H
51EØ  2Ø F Ø             BRA      POWER
51E2  Ø1          OUT   NOP
51E3  39                RTS
51E4                    END
```

STATEMENTS =1184

FREE BYTES =11986

NO ERRORS DETECTED
--

```
                        *
                        *       ASSEMBLY ROUTINE FOR
                        *       PSD MESUREMENT
                        *
49CA                     XREAL   EQU     $49CA
4DCA                    XIMAG   EQU     $4DCA
4BCA                    XREAL1  EQU     $4BCA
4FCA                    XIMAG1  EQU     $4FCA
5600                    XRESUL  EQU     $5600
425F                    DPMUL   EQU     $425F
4259                    H       EQU     $4259
425A                    L       EQU     $425A
425B                    D       EQU     $425B
425C                    E       EQU     $425C
                        *
                        *       SUBROUTINE OVERFLOW
                        *
5400                            ORG     $5400
5400  01                OFL     NOP
5401  78 42 5A                  ASL     L
5404  79 42 59                  ROL     H
5407  78 42 5A                  ASL     L
540A  79 42 59                  ROL     H
540D  78 42 5A                  ASL     L
5410  79 42 59                  ROL     H
5413  78 42 5A                  ASL     L
5416  79 42 59                  ROL     H
5419  39                        RTS
                        *
                        *       MAIN PSD ROUTINE STARTS
                        *
5500                            ORG     $5500
5500  00 00             TEMP    FDB     $0000
5502  00 00             TEMP1   FDB     $0000
5504  00 00             TEMP2   FDB     $0000
5506  00 00             TEMP3   FDB     $0000
5508  00 00             TEMP4   FDB     $0000
550A  CE 4B CA                  LDX     #XREAL1
550D  FF 55 00                  STX     TEMP
5510  CE 4F CA                  LDX     #XIMAG1
5513  FF 55 02                  STX     TEMP1
5516  CE 56 00                  LDX     #XRESUL
5519  FF 55 08                  STX     TEMP4
551C  FE 55 00          INNER   LDX     TEMP
551F  A6 00             START   LDA A   ,X
5521  08                        INX
5522  E6 00                     LDA B   ,X
5524  08                        INX
5525  FF 55 00                  STX     TEMP
5528  B7 42 59                  STA A   H
552B  F7 42 5A                  STA B   L
552E  BD 54 00                  JSR     OFL
5531  FE 42 59                  LDX     H
5534  FF 42 5B                  STX     D
                        *
                        *       DPMUL IS 16BIT MULTIPLY
```

```
                        *       ROUTINE IN FFT ROUTINE
                        *
5537  BD 42 5F          JSR     DPMUL
553A  FE 42 59          LD..    H
553D  FF 55 04          STX     TEMP2
5540  FE 55 02          LDX     TEMP1
5543  A6 00             LDA A   ,X
5545  08                INX
5546  E6 00             LDA B   ,X
5548  08                INX
5549  FF 55 02          STX     TEMP1
554C  B7 42 59          STA A   H
554F  F7  42 5A          STA B   L
5552  BD 54 00          JSR     OFL
5555  FE 42 59          LDX     H
5558  FF 42 5B          STX     D
555B  BD 42 5F          JSR     DPMUL
555E  FE 42 59          LDX     H
5561  FF 55 06          STX     TEMP3
5564  B6 55 05          LDA A   TEMP2+1
5567  F6 55 04          LDA B   TEMP2
556A  BB 55 07          ADD A   TEMP3+1
556D  F9 55 06          ADC B   TEMP3
5570  FE 55 08          LDX      TEMP4
5573  E7 00             STA B   ,X
5575  08                INX
5576  A7 00             STA A   ,X
5578  08                INX
5579  FF 55 08          STX     TEMP4
557C  FE 55 00          LDX     TEMP
557F  8C 4B CA          CPX     #$4BCA
5582  27 17             BEQ     END1
5584  8C 4D CA          CPX     #$4DCA
5587  27 03             BEQ     NEXT
5589  7E 55 1F          JMP     START
558C  CE 49 CA  NEXT    LDX     #XREAL
558F  FF 55 00          STX     TEMP
5592  CE 4D CA          LDX     #XIMAG
5595  FF 55 02          STX     TEMP1
5598  7E 55 1C          JMP     INNER
559B  01        END1    NOP
559C                    END

STATEMENTS =98

FREE BYTES =24679

NO ERRORS DETECTED
--
```